Detection of Aedes Aegypti Mosquito by Digital Image Processing techniques and Support Vector Machine

by

Anna Monica De Los Reyes
Anna Camille Ramirez

A Thesis Report Submitted to the School of Electrical, Electronics, and Computer Engineering in Partial Fulfilment of the Requirements of the Degree

Bachelor of Science in Computer Engineering

Mapúa Institute of Technology

December 2015

ACKNOWLEDGEMENT

This Thesis Report would not have been successful without the existence of the following people,

To our parents, who supported us financially and who believed and continually encourage us in finishing this report.

To our Advisers, Engr. Jumelyn L. Torres and Dionis A. Padilla, who continuously helped us in giving their support to pursue this paper, and who shared their depth knowledge to improve our thesis report.

To our friends, who shared some of their knowledge and who continuously motivate us that we can able to pass everything that we need to finish in this report.

To the staffs of RITM (Research Institute of Tropical Medicine), who helped us by allowing us to work in their facilities, and even gave us ideas on how we can work to our thesis.

Lastly, to the Man who never left us in order to finish this work, who gave us the ability to make things work for our thesis, who gave us more than enough for us to do these things, God.

APPROVAL SHEET

This is to certify that we have supervised the preparation of and read the thesis paper prepared by **Anna Monica M. De Los Reyes** and **Anna Camille A. Ramirez** entitled **Detection of Aedes Aegypti Mosquito by Digital Image Processing Techniques and Support Vector Machine** and that the said paper has been submitted for final examination by the Oral Examination Committee.

Jumelyn L. Torres
Thesis Adviser

Dionis A. Padilla
Thesis Adviser

As members of the Oral Examination Committee, we certify that we have examined this paper, presented before the committee on **November 21, 2015**, and hereby recommended that it be accepted in fulfilment of the thesis requirements for the degree **Bachelor of Science in Computer Engineering**.

Joshua B. Cuesta
Panel Member

Jose B. Lazaro
Panel Member

Carlos C. Hortinela IV
Committee Chair

This thesis paper is hereby approved and accepted by the School of Electrical, Electronics, and Computer Engineering in partial fulfilment of the requirements for the degree **Bachelor of Science in Computer Engineering.**

Alejandro H. Ballado, Jr.
Dean, School of EECE

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

In this paper, it uses digital image processing and support vector machine (SVM) to detect aedes aegypti mosquito. Aedes aegypti mosquito is a carrier of dengue virus which causes the problem that is called dengue hemorrhagic fever. As of today there is still no particular medicine for the treatment of such. As a result the study proposed a way on how to detect if a mosquito is an aedes aegypti or not using SVM in Matlab. First, it was trained by feeding images of aedes aegypti and not aedes aegypti such as aedes albopictus and culex that were taken at different views, highlighting their unique features. In the image processing, the features were extracted and then classified to aedes aegypti or not aedes aegypti. Confusion matrix was used to show the accuracy of the system in detecting the Aedes Aegypti mosquito.

Keywords: aedes aegypti, image processing, support vector machine, confusion matrix, matlab

Chapter 1

INTRODUCTION

Aedes Aegypti mosquito is a carrier of dengue virus which causes an illness like severe-flue, and at times a potentially serious sickness problem that is called dengue hemorrhagic fever. As of today, there's still no particular medicine for the treatment of dengue fever or any vaccine invented to be immune from the bite of a dengue carrier mosquito. In result of this, still, a huge population of people anywhere in the world is being affected and has died from the dengue virus. People will know that a place is infected by dengue carrier mosquito when news spread that a person was already taken to the hospital. The World Health Organization (WHO) said that dengue is the "fastest emerging arboviral infection" in the world. Dengue fever has greatly affected the world's population; partially more are in the region of Asia-Pacific. Dengue is transferred when a human is bitten by infected Aedes Aegypti female mosquito. Dengue cannot be transmitted directly from human to human which means a mosquito is necessary for transmission of the dengue virus. There is no way to end the Aedes Aegypti mosquito from spreading dengue virus besides from the diligent usage of insect repellent and using mosquito nets. One study detects mosquito from its behavior, characteristics and territorial which the result could be important to fight against malaria, yet the study does not determine if the detected mosquito is an Aedes Aegypti which is a dengue carrier mosquito.

In order to reduce the number of people infected by the dengue virus, different techniques in detecting mosquitoes have been developed. In the study of Jahangir et al, they used an image processing in detecting mosquitoes and their locations. They found that mosquitoes can be detected and differentiated by their physical, territorial and behavioral patterns through these methodologies. In order to classify that the detected insect is a mosquito, the study includes flies and bees for comparison in terms of size, body, shape, color, antennae, hind legs and shape parameters. In another study by Monette Vessell, the researcher developed a system that creates a trap that can detect mosquitoes that would detect mosquitoes entering the trap and transmit the census data from a remote location to a public health agency. The study was developed for monitoring mosquito populations. The researcher stated that in monitoring populations of mosquito can provide early detection of potential epidemics and can minimize the amount of chemicals applied to control mosquito.

As of this year, many studies were designed in able to supply information in detecting mosquitoes. A major factor in influencing the studies is that, there is no specific medicine or vaccine that can treat a person that has a dengue virus, thus it ends in numerous dengue cases in the country or in the Asia-Pacific Region. Since, detecting an Aedes Aegypti mosquito is a different thing to be implemented, then this study deals with the problem that the existing devices only identifies if a detected insect is a mosquito or not and monitors the mosquito population.

The main aim of this study is to detect and determine if the captured mosquito is an Aedes Aegypti or not basing from its physical features in order to develop a device that could help in dengue outbreak cases. Specific objectives of the study are as follows: (1) to design a prototype that can capture an image of Aedes Aegypti mosquito and indicate its physical features (2) to classify mosquito using Support Vector Machine or SVM (3) to match the captured image of mosquito from an image of an Aedes Aegypti mosquito using digital image processing techniques in MATLAB.

The study can be a contributing factor to other laboratories in examining the Aedes mosquito. And also it can be a reference for further researches related to dengue problem. This may also supply data for image processing, especially for a very small species such as mosquitoes. Moreover, this study shows one of the applications of support vector machine in classification purposes.

This study focuses on the detection of an Aedes Aegypti together in measuring the accuracy or exactness of the characteristics of the mosquito. This will have two parts, namely: software and hardware development. For software part, it will be the image that must be processed in order to match or identify the kind of mosquito that has been captured. The hardware part will consist of a microscopic camera with a specification of 500x optical zoom or 5x digital zoom in order to have a better quality in capturing the mosquito. The camera in this study uses manual focus on the subject because the camera was not compatible for the auto focus on the Matlab application. However, the mosquito must only be captured in a "not moving state" and be given from a closed environment. At the same time, the background of the image must be plain light colors.The resulting time for the process will not be calculated. The comparison or matching of mosquitoes will only focus from 3 types: Aedes Aegypti, Aedes Albopictus and Culex, where Aedes Albopictus and Culex will be determined as “Not Aedes Aegypti”. The comparison of the mosquitoes will based from five unique features of an Aedes Aegypti mosquito namely proboscis, pulps, clypeus, scutum and wingscales. The study’s system must be updated every after training of SVM and the destination folder for the classification data and images must be correct for it not to malfunction or to have an error when detecting mosquito images.

Chapter 2

REVIEW OF RELATED LITERATURE

Mosquito

Mosquitoes are a family of small, midge-like flies. They can be male or female. Those male mosquitoes drink only fluids that are sugary as they can find; they are not those bloodsuckers. But the female mosquitoes are blood-eating pests. They create such tremendous impact on a human’s life because of transmitting harmful diseases, more specifically a dengue virus. The female mosquito that carries dengue can be identified in this type:(CDC, 2012)

Figure 2.1 Aedes Aegypti (CDC, 2012)

Aedes Aegypti referring to Figure 2.1 is also known as “Egyptian tiger mosquito”. It is a small, dark mosquito with a lyre shaped markings and banded legs. They bite primarily during the day. (CDC, 2012)

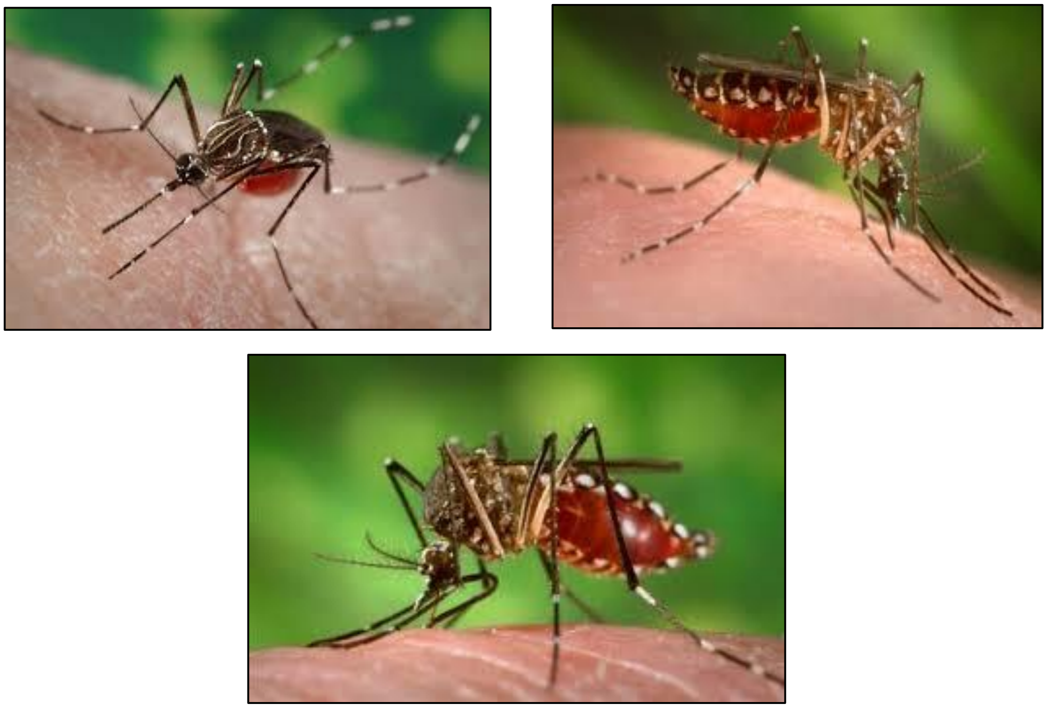


Figure 2.2 Three-D Images of Aedes Aegypti (CDC, 2012)

From Figure 2.2 are the sample 3D images of an aedes aegypti mosquito that will be used for the training of the SVM tool.

Species	Head	Thorax	Abdomen
Aedes aegypti	Proboscis: Dark; Pulps: Tipped with silvery-white; Clypeus: White scales.	Wings Scales dark; Legs color: Tarsal segment 5 is entirely white; White basal bands 2*6 for Hind legs and 3 for mid & forelegs.	Scutum: Lyre shaped, White scales.
Aedes albopictus (Stegomyia albopicta)	Pulps: Tipped with silvery-white; Clypeus: Black.	Wings Scales dark; Legs color: Tarsal segment 5 is entirely white; White basal bands 2*6 for Hind legs and 3 for mid & forelegs; Thorax: Sides are many silvery-white.	Scutum: One middle silvery-white stripe down.
Aedes vexans	Proboscis: Dark, Pulps: Dark.	Wings Scales dark; Legs color: Tarsal segment is entirely narrow white; narrow White basal bands 2*6 for Hind legs and 3 for mid & forelegs.	Shape: V, Color: Pale.
Culex territans	Proboscis: Dark, Pulps: Dark	Wings Dark, narrow; Legs color: Dark.	Color: Narrow APICAL bands.
Culex restuans	Proboscis: Dark; Edge is light white, Pulps: Dark.	Wings Dark, narrow; Legs color: Dark; some white spots; Thorax: Patches; pale scales.	Dark & white basal bands, scutum: Copper color; rarely 2 pale spots.
Deinocerites cancer	Proboscis: Dark; Antenna longer than proboscis, Pulps: Dark.	Wings Dark, narrow; Legs color: Dark.	Color: Copper brown.
Ochlerotatus bahamensis	Proboscis: Dark, Pulps: White tipped.	Wings Dark, narrow; Legs color: Hind legs dark with White basal bands.	Sputum: lines of golden & white scales.
Ochlerotatus infirmatus	Proboscis: Dark; Pulps: Dark.	Wings Dark, narrow; Legs color: Dark.	Color: Dark scale with basal;
Ochlerotatus triseriatus	Proboscis: Dark; Pulps: Dark.	Wings Dark; narrow; Legs color: Dark; Thorax: patches of silver-white scales.	Scutum: Dark; Legs are white of edge.

Figure 2.3 Characteristics of Different types of Mosquito (Jahangir, 2014)

Most of the mosquitoes have angular shaped bodies with an abdomen part, thorax and head. The legs are long, the abdomen is slender and narrow with dark color, though some of them have white circular lines. Their hind legs are both light and dark in color and are thin. It is also stated that the minimum length of mosquitoes is about 16mm which is very important for pattern recognition design.

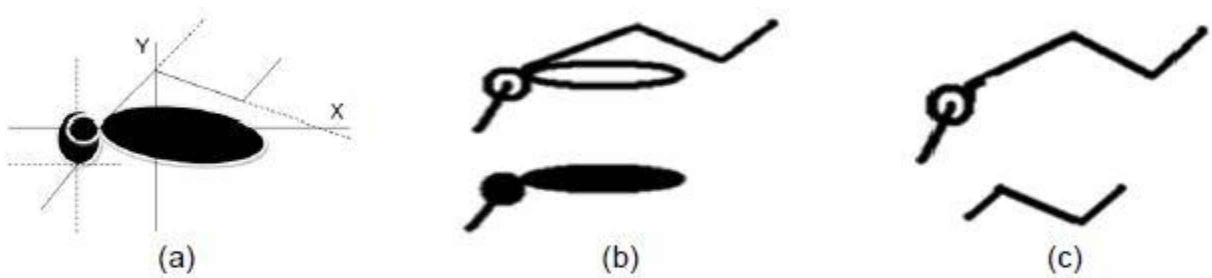


Figure 2.4 (a) model of mosquito; (b) shape of mosquito; (c) shape of mosquito with hind leg

From the figure shown above Figure 2.4, the angle between the head and abdomen is approximately ±135°. Hence the body shape is live ‘V or Λ’ but if it is viewed with the addition of the hind legs, then the pattern is more like ‘Z’. (Jahangir, 2014)

MATLAB

It has been said that a system of digital image processing is typically created around a computer system having hardware for an image and for the video acquisition, storage, and even the display. There is a software part of the system that normally consists of modules that can perform different specific tasks (Gonzales, 2009). MATLAB together with its toolbox which is the Image Processing is one of the choices of software for image processing.

Based from the book Digital Image Processing Using MATLAB by Gonzales, MATLAB brings an extensive set of functions that can process multidimensional arrays to digital image processing. In MATLAB software there is a specific toolbox that is used for image processing. This Image Processing Toolbox has a collection of functions that extend the numeric computing environment in MATLAB. Because of these built-in functions image processing operations becomes easy to write in compact, clear manner, therefore offers an ideal environment of software prototyping to solve the problems of image processing (Gonzales, 2009).

Image Processing

It was stated that the technique image processing was used to enhance the raw images that was taken from cameras that is placed on satellites and may be on aircrafts or just images taken in handy camera that most people have this days. It was also said that during the last 4 to 5 decades, different kinds of techniques have been developed in image processing. This process has become popular due to easy availability of powerful personnel computers, large size memory devices, graphics software etc. (Rao, 2006).

According to Rao, in image processing, analog and digital are two methods. Digital image processing has a definition of being as one of the subject of numerical representations of objects in a sequence of operations in order to obtain a preferred result. It is starting from by having one image and generates a modified or new version of the same. Therefore, it is a process that gets a picture or image into another (Rao, 2006). Image processing has various techniques like enhancing the image, image preprocessing, and image restoration, reconstruction of images, image data compression, image representation and image analysis. Noise filtering and histogram modification are both image enhancement and segmentation is an image analysis.

Segmentation

For image segmentation, Young stated in 2007 that it is essential to distinguish the object of interest from the rest in analyzing objects in images. When says ‘rest ’, it also referred to as the background. The segmentation techniques that separate the background from the object of interest are usually called as segmenting the background from the foreground. According to the previous study which entitled Characteristics Analysis and Detection Algorithm of Mosquitoes by Jahangir, et. Al, the background subtraction method is effective to detect for slow motion flying mosquito. The pixel value for the background images and the detected moving images are not the same so difference between these two values were defined as the region of interest for mosquito detection. In differentiating the modeling and updating, the said two pixel values are very important. In their study, they used the algorithm of Adaptive Surendra (ASA) for background estimation. Then for the detection of flying mosquito the researcher used the Inter Frame Differencing Algorithm (IFDA). For their mosquito detection, Young et. Al prioritized frame differencing, background extraction and updating, background subtraction and motion detection. The process used by Young in 2007 is shown in Figure 2.5. The background of the image has been subtracted by thresholding and region growing method so that it can be detected as quickly as possible.

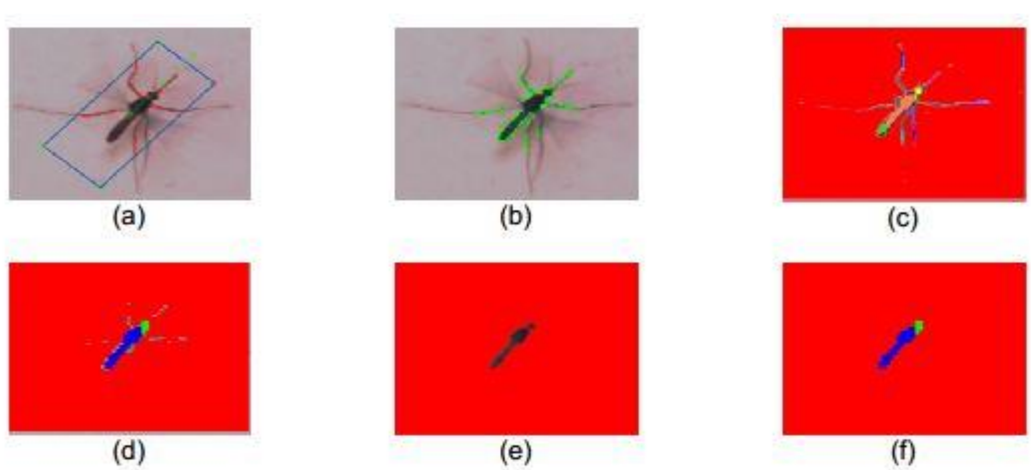


Figure 2.5 Background Segmentation for Mosquito Detection (Young, 2007)

Noise Filtering

Rao stated that noise filtering is used to filter the information that is unnecessary from an image and is used to remove from the images the different types of noises. This feature is mostly interactive. Various filters like high pass, low pass, median, mean etc., are available (Young, 2007). In previous study entitled Characteristics Analysis and Detection Algorithm of Mosquitoes by Jahangir et Al, they used image filtering to eliminate the noise of an image. This process is important to get the efficient pixel value. They developed the program which is noise extraction and rejection through MATLAB (Jahangir, 2014). In another study which is Monitoring of Pest Insect Traps Using Image SensorsandDspic by ThulasiPriya, et. Al, they used the process of denoising concept on MATLAB. The input image, which is captured by camera, is in RGB format and converted the input image (Figure 2.6) to gray scale image Figure 2.7) to identify the insects from back ground. Their next step was to the noise from the image. They encounter difficulty when the noise has not been removed from the image because it has high frequency that is difficult in finding or detecting the edge (Priya, 2013).

Figure 2.6 Sample Input image (Priya, 2013)

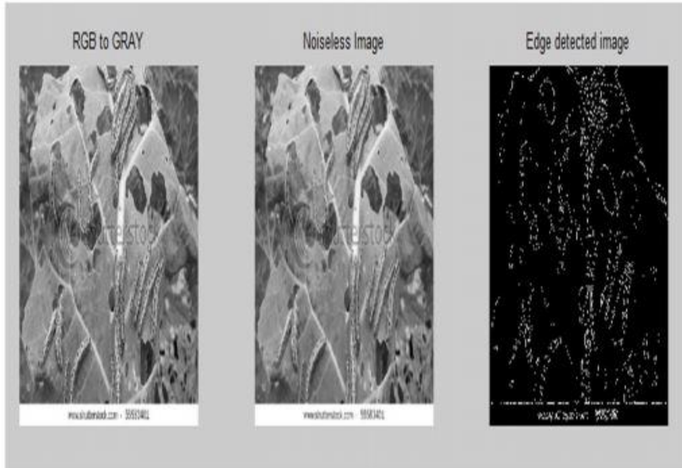


Figure 2.7 Simulation result (Priya, 2013)

Histogram Modification

It was stated that histogram is in great importance in image processing. It reflects the characteristics of an image. The characteristics of image can be modified by modifying the histogram. An example of this is histogram equalization. Histogram equalization is defined as a nonlinear stretch that has the ability to redistribute pixel values to have the same number of pixels, approximately, where every value is within the range. The result approximates of having a flat histogram. Therefore, the contrast increases at the peaks and decreases at the tails (Rao, 2007). In previous study, Characteristics Analysis and Detection Algorithm of Mosquitoes by Jahangir et. Al, they used the histogram method to produce the area, mean, median, and standard deviation etc. of the image processing data which was collected on mosquitoes. To analyze insect size, number of objects, proboscis, body, color, antennae and shapes, the histogram was applied.

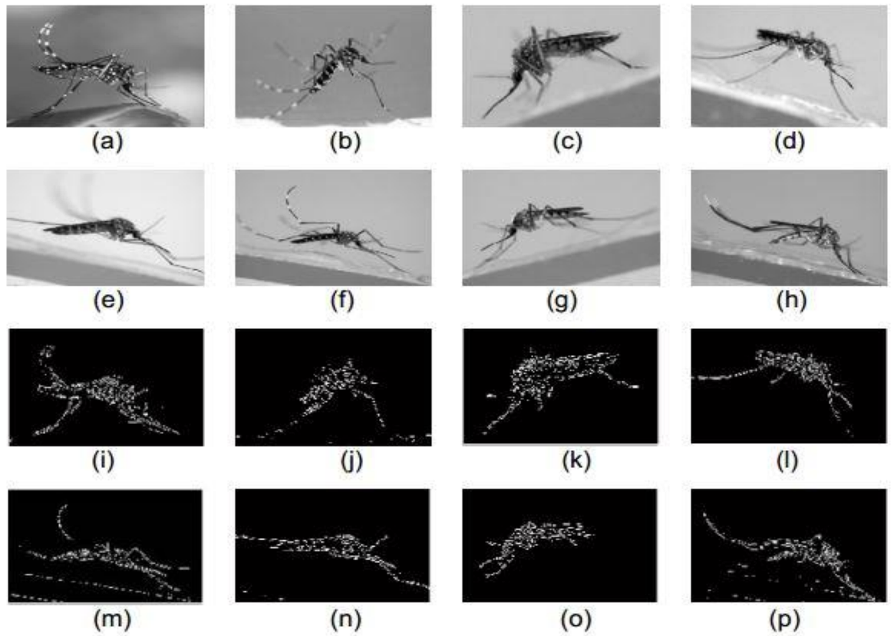


Figure 2.8 Mosquitoes pattern images (Jahangir, 2014)

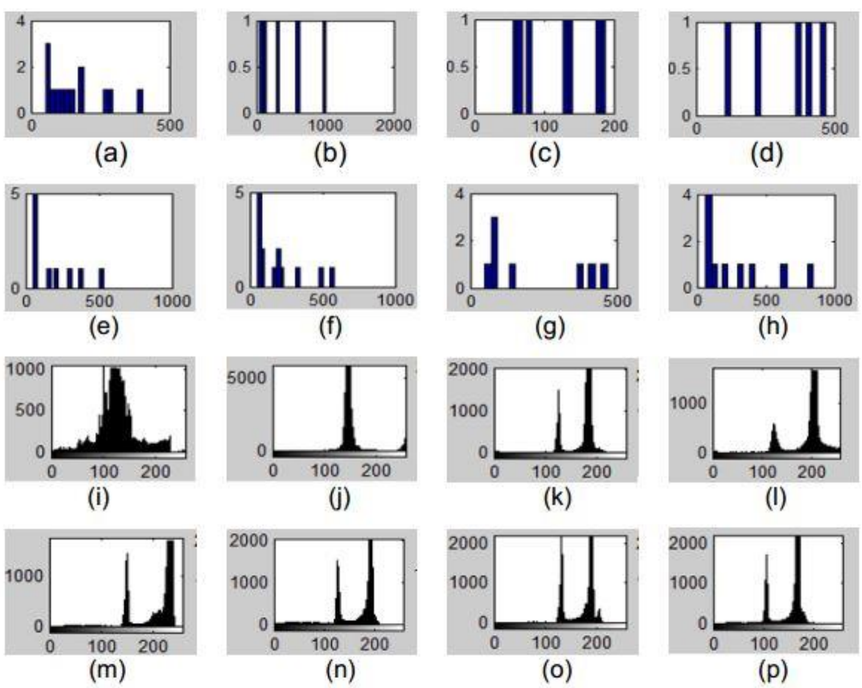


Figure 2.9 Statistics of Mosquitoes; (a)-(h) histogram, and (i)-(p) area histogram (Jahangir, 2014)

In figure 2.8 and figure 2.9 from the study of Jahangir et. Al, the mosquito's positional pattern was observed and the Edge pattern of the mosquito was detected by MATLAB Edge processing. The results show the mosquito's positional histogram. In their process of imaging, they can detect and determine where the mosquito is and they can even predict where it is about to go based on analyzing the patterns as to where the mosquito will likely be in the future as shown in Figure 2.8(i)-(p). In their study, they have three species or insects to compare which are mosquitoes, bees and flies. The histogram analysis showed that the mosquito's territory is smaller than that of flies and bees. This can be seen in Figure 2.9.Based on the figures of territorial edge patterns, the researcher concluded that a mosquito's territorial pattern shape is narrow and long where the pattern for flies is almost triangular or 'V' or 'Z' or 'Curve' or 'Λ' in shape as in Figure 2.8(i)-(p) (Jahangir, 2014).

Feature Extraction

Mundada used feature extraction method in their study to detect presence of pests. They consider or include some properties like the region properties and the property, gray covariance matrix. From the extraction of an image, properties like the mean, standard deviation, the entropy, the energy, contrast, the correlation and eccentricity are considered. After extraction of the image, the properties were compared and the support vector machine is trained and used for classifying the images. Support Vector Machines (SVM's) are rather a new method to learn where it is used for classifying binary (Mundada, 2013).

In Mundada's method or process in doing their study, the detection method was after feature extraction method. In extraction method, the support vector machine receives the input image. The support vector machine used was trained from the collected data from their database. The input for the support vector machine that the researcher has trained was the features of the input image that were extracted. From comparing the parameters of the database, the support vector machine produces the output. After detection the system, then identified the pests. When the finding of the leaf is infected, then their next step is to find the next. Also, if the leaf has a finding to be infected, then its next step is finding what type of pest (Mundada, 2013).

Figure 2.10 Database for classification (Mundada, 2013)

The researcher classified the pests into 2 categories which are named as, whiteflies and aphids. For identification, there is a mask that is a special type that is used after averaging filtering. Then the image that is filtered is convolved with the mask. Then from the process of extraction for the property region and the property of gray co-occurrence matrix, classifying is completed in two types, which are whiteflies and aphids. They also considered region properties similar to standard deviation and contrast for the identification. For choosing the category the researcher used again the SVM classifier. Shown above in Figure 2.10 (Mundada, 2013) is the database that is –given for the training of second SVM.

SIFT and RANSAC Algorithm

In another research study conducted by Qiuet. Al., the researcher uses the Harris corner detection and SIFT algorithm (Scale Invariant Feature Transform) to extract distinct features of an image. The researcher proposed to use the algorithm for their research mainly for detecting images that have been scaled and rotated differently from an original image. The SIFT algorithm is the one to identify and define, measure invariant and rotation invariant local features in images. Another algorithm used by another researcher is the RANSAC. RANSAC is random sample consensus which approximates calculated model from records that contains outliers. RANSAC basically used to find cohesion amongst two sets of points for feature detection.

Image Texture Analysis

In the paper of Bergman et. al., titled Detection of Texture Areas in Images Using a Disorganization Indicator Based on Component Counts, the researcher presented an algorithm for the digital image recognition of textured areas. The texture detection used by the researcher is for the analysis of arrangement of local areas of image. Moreover, the texture detection for the paper separates between noise edges and texture (Bergman, 2007). Equations below were used for determining the different properties:

Energy:

$$f_1 = \sum_i \sum_j p(i, j)^2. \qquad (2.1)$$

Contrast:

$$f_2 = \sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i, j) \middle| |i-j| = n \right\}. \qquad (2.2)$$

Homogeneity:

$$f_4 = \sum_i \sum_j \frac{1}{1+(i-j)^2} p(i, j). \qquad (2.3)$$

Correlation: (2.4)

$$f_3 \frac{\sum_i \sum_j (ij)p(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y}.$$

Entropy: (2.5)

$$f_5 = -\sum_i \sum_j p(i,j) \log(p(i,j)).$$

Dissimilarity:

$$f_7 = \sum_i \sum_j |i-j| \cdot p(i,j). \tag{2.6}$$

Maximum Probability:

$$f_{10} = \underset{i,j}{\mathrm{MAX}}\, p(i,j). \tag{2.7}$$

Support Vector Machine

Support Vector Machine is a supervised learning machine that is used for classification analysis. SVM has its learning algorithm for it to be able to identify and analyze patterns. Its main concept is about hyperplane which separates different classification data for different objects. The hyperplane is also defined as the decision boundary for SVM. The figure below shows how the classification of the SVM. The 'x' and 'o' are the different data for an object and the line drawn between them is the hyperplane.

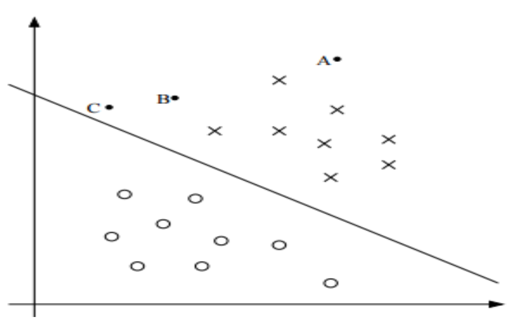


Figure 2.11 SVM graphical representation (Andrew Ng)

Chapter 3

DETECTION OF AEDES AEGYPTI MOSQUITO USING DIGITAL IMAGE PROCESSING TECHNIQUES AND SUPPORT VECTOR MACHINE

Abstract

In this paper, it uses digital image processing and support vector machine (SVM) to detect aedes aegypti mosquito. Aedes aegypti mosquito is a carrier of dengue virus which causes the problem that is called dengue hemorrhagic fever. As of today there is still no particular medicine for the treatment of such. As a result the study proposed a way on how to detect if a mosquito is an aedes aegypti or not using SVM in Matlab. First, it was trained by feeding images of aedes aegypti and not aedes aegypti such as aedes albopictus and culex that were taken at different views, highlighting their unique features. In the image processing, the features were extracted and then classified to aedes aegypti or not aedes aegypti. Confusion matrix was used to show the accuracy of the system in detecting the Aedes Aegypti mosquito.

Keywords: aedes aegypti, image processing, support vector machine, confusion matrix, matlab

Introduction

Aedes Aegypti mosquito is a carrier of dengue virus which causes an illness like severe-flue, and at times a potentially serious sickness problem that is called dengue hemorrhagic fever. As of today, there's still no particular medicine for the treatment of dengue fever or any vaccine invented to be immune from the bite of a dengue carrier mosquito. In result of this, still, a huge population of people anywhere in the world is being affected and has died from the dengue virus. People will know that a place is infected by dengue carrier mosquito when news spread that a person was already taken to the hospital. The World Health Organization (WHO) said that dengue is the "fastest emerging arboviral infection" in the world. Dengue fever has greatly affected the world's population; partially more are in the region of Asia-Pacific. Dengue is transferred when a human is bitten by infected Aedes Aegypti female mosquito. Dengue cannot be transmitted directly from human to human which means a mosquito is necessary for transmission of the dengue virus. There is no way to end the Aedes Aegypti mosquito from spreading dengue virus besides from the diligent usage of insect repellent and using mosquito nets. One study detects mosquito from its behavior, characteristics and territorial which the result could be important to fight against malaria, yet the study does not determine if the detected mosquito is an Aedes Aegypti which is a dengue carrier mosquito.

In order to reduce the number of people infected by the dengue virus, different techniques in detecting mosquitoes have been developed. In the study of Jahangir et al, they used an image processing in detecting mosquitoes and their locations. They found that mosquitoes can be detected and differentiated by their physical, territorial and behavioral patterns through these methodologies. In order to classify that the detected insect is a mosquito, the study includes flies and bees for comparison in terms of size, body, shape, color, antennae, hind legs and shape parameters. In another study by Monette Vessell, the researcher developed a system that creates a trap that can detect mosquitoes that would detect mosquitoes entering the trap and transmit the census data from a remote location to a public health agency. The study was developed for monitoring mosquito populations. The researcher stated that in monitoring populations of mosquito can provide early detection of potential epidemics and can minimize the amount of chemicals applied to control mosquito.

As of this year, many studies were designed in able to supply information in detecting mosquitoes. A major factor in influencing the studies is that, there is no specific medicine or vaccine that can treat a person that has a dengue virus, thus it ends in numerous dengue cases in the country or in the Asia-Pacific Region. Since, detecting an Aedes Aegypti mosquito is a different thing to be implemented, then this study deals with the problem that the existing devices only identifies if a detected insect is a mosquito or not and monitors the mosquito population.

The main aim of this study is to detect and determine if the captured mosquito is an Aedes Aegypti or not basing from its physical features in order to develop a device that could help in dengue outbreak cases. Specific objectives of the study are as follows: (1) to design a prototype that can capture an image of Aedes Aegypti mosquito and indicate its physical features (2) to classify mosquito using Support Vector Machine or SVM (3) to match the captured image of mosquito from an image of an Aedes Aegypti mosquito using digital image processing techniques in MATLAB.

The study can be a contributing factor to other laboratories in examining the Aedes mosquito. And also it can be a reference for further researches related to dengue problem. This may also supply data for image processing, especially for a very small species such as mosquitoes. Moreover, this study shows one of the applications of support vector machine in classification purposes.

This study focuses on the detection of an Aedes Aegypti together in measuring the accuracy or exactness of the characteristics of the mosquito. This will have two parts, namely: software and hardware development. For software part, it will be the image that must be processed in order to match or identify the kind of mosquito that has been captured. The hardware part will consist of a microscopic camera with a specification of 500x optical zoom or 5x digital zoom in order to have a better quality in capturing the mosquito. The camera in this study uses manual focus on the subject because the camera was not compatible for the auto focus on the Matlab application. However, the mosquito must only be captured in a "not moving state" and be given from a closed environment. At the same time, the background of the image must be plain light colors.The resulting time for the process will not be calculated. The comparison or matching of mosquitoes will only focus from 3 types: Aedes Aegypti, Aedes Albopictus and Culex, where Aedes Albopictus and Culex will be determined as “Not Aedes Aegypti”. The comparison of the mosquitoes will based from five unique features of an Aedes Aegypti mosquito namely proboscis, pulps, clypeus, scutum and wingscales. The study’s system must be updated every after training of SVM and the destination folder for the classification data and images must be correct for it not to malfunction or to have an error when detecting mosquito images.

Methodology

Conceptual Framework

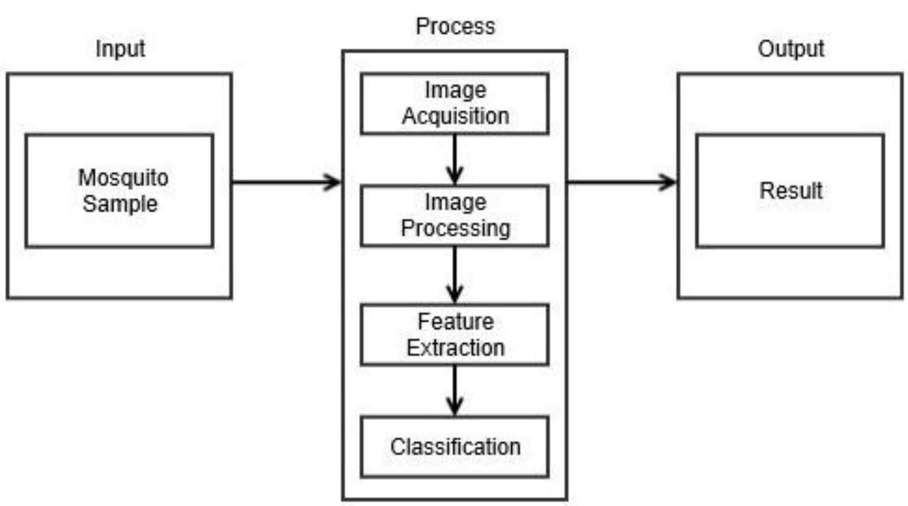


Figure 3.1 Conceptual Framework

Based from Figure 3.1, the system starts by getting a mosquito given in a closed environment. Once the mosquitoes are captured by a camera, the images are transferred and saved in a specified format (jpeg, bmp, png). The image is processed using MATLAB software, where segmentation, noise filtering takes place. After using the processes, the features of the mosquito are extracted, then data that has been gathered is given to SVM that lets the system display a message on the program telling that the mosquito is a type of an Aedes Aegypti mosquito or not.

Data Flow Diagram

Figure 3.2 Context Diagram

Figure 3.2 shows the context diagram for aedes aegypti detection system. There is only one entity for the system which is the User. The user provides the image that is needed by the system. The system then processes the image, and then the classified image is the result or output of the system that was needed by the user.

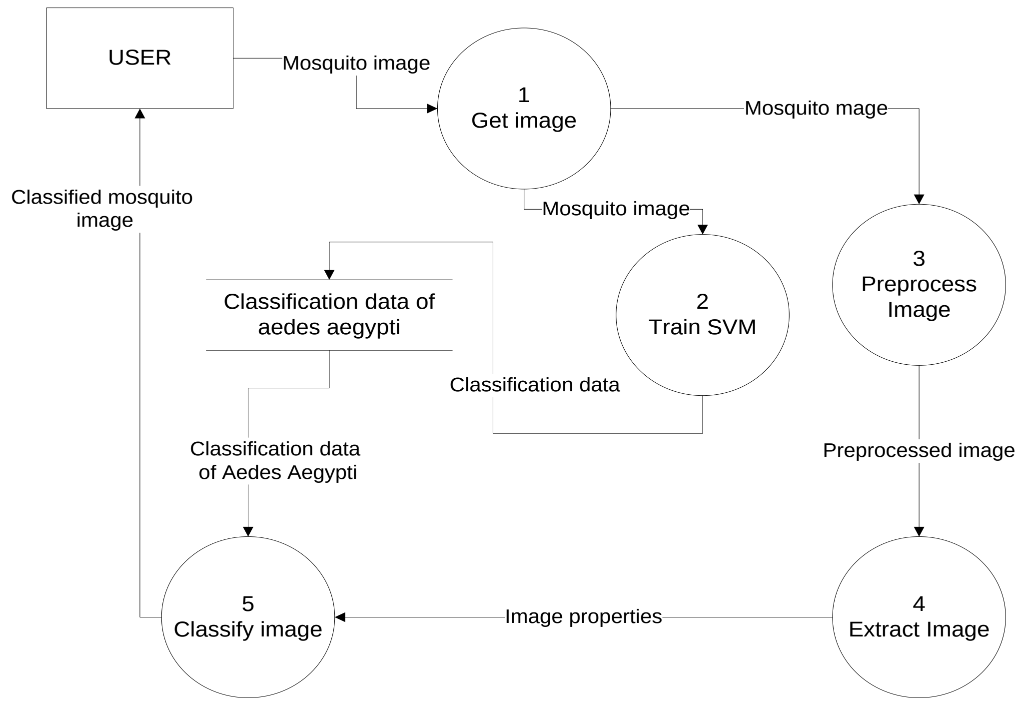


Figure 3.3 Diagram 0 for the system

Figure 3.3 shows the more elaborate processes of the system. The user provides the image for the system then the process Get image let user to browse or capture an image of mosquito. There is option for training the SVM where images of aedes aegypti mosquito were given to the system and after training the classification data of aedes aegypti was stored. After that, the image undergoes preprocessing which includes background segmentation and noise filtering. The preprocessed image then was extracted. After extraction of image, the image properties that were produced by the process and the classification data of aedes aegypti that was stored when training was used for the classification of the image. After the classify image processor, the classified image, whether it is an aedes aegypti or not was displayed. The following figure refers to the flowchart of every process in the diagram 0 of DFD.

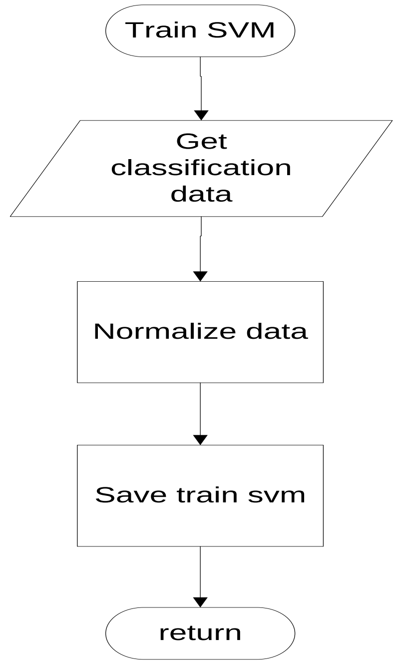


Figure 3.4 Train SVM Flowchart

Figure 3.4 shows how the system trains its SVM. First, it gets the classification data which comes from the image properties that was saved on the classificationdata.xlsx. The classification data that have values more than 1 is then normalized. After normalizing, the SVM is trained using the data and then saved.

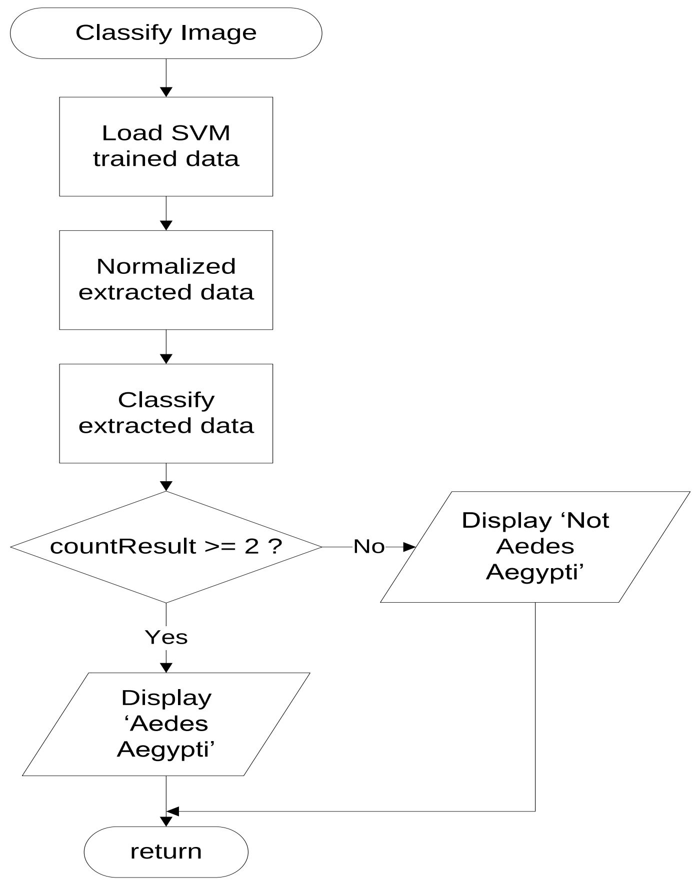


Figure 3.5 Classify Image Flowchart

Figure 3.5 shows the flow of how the system classifies mosquito image. First, it loads the classification data of the trained images. Then the extracted image properties are normalized. After that, the SVM classifies if the new image is an Aedes Aegypti or not based on its image properties. The countResult is for when the SVM classifies the image two times or more than two, the image is an Aedes Aegypti or not.

System Flow Chart

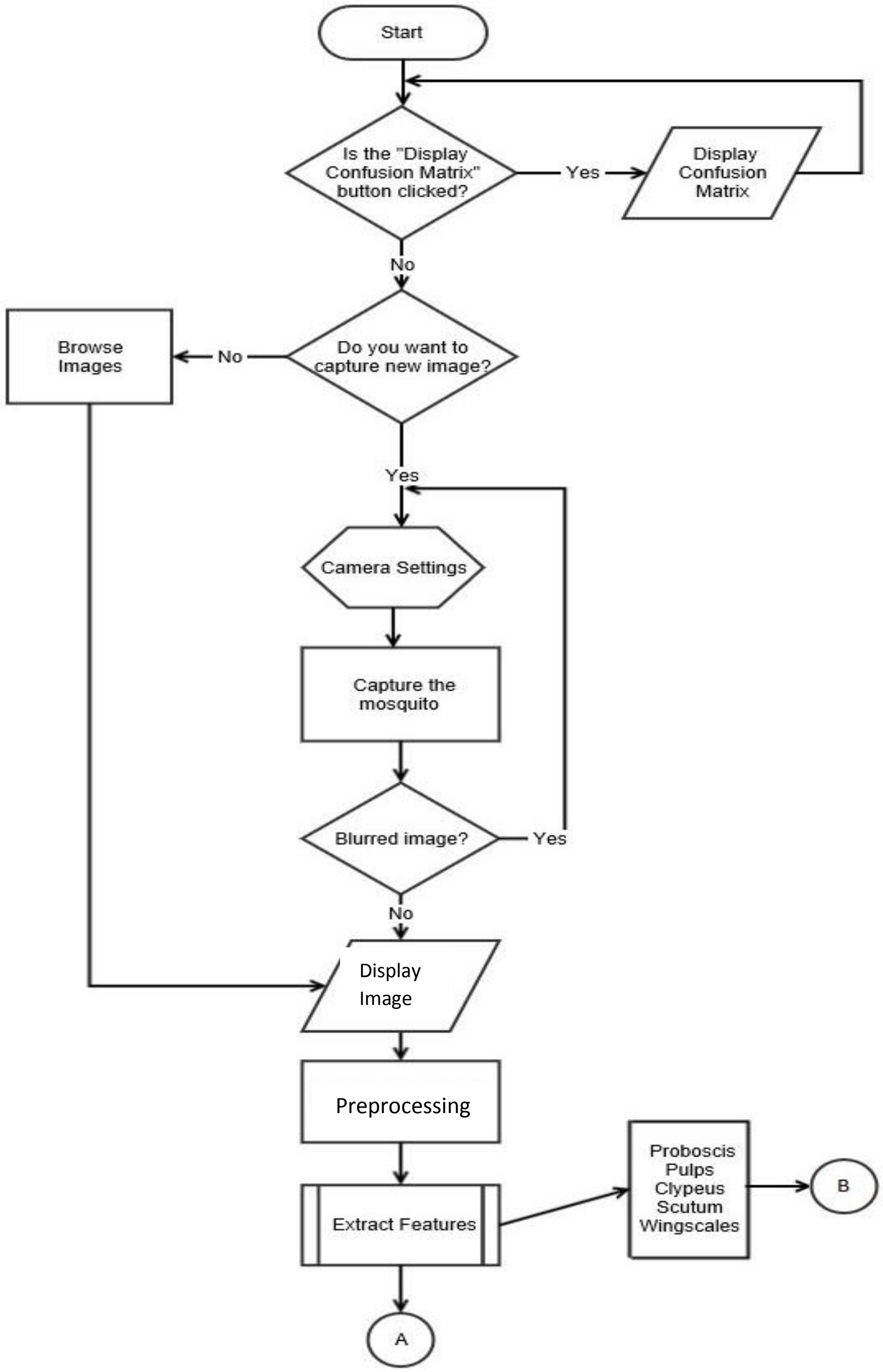


Figure 3.6.1 System Flowchart

From Figure 3.6.1 it shows how the system flows. User may choose whether to capture an image or check the confusion matrix. The camera settings include the tilt and pan of the camera which was controlled by two potentiometers were connected to a microcontroller. Once the user was satisfied with the view of the object to be captured, which was the mosquito, the system provides an interface that captures an image without having to click or touch the camera setup. From the captured image, the preprocessing takes place. It consists of techniques of background segmentation and noise filtering; from background segmentation the object of interest which was the mosquito was isolated from the background. From segmenting the background, removal of noise from the captured image was done. Noise filtering helped improve the determining of the features of the captured image of the mosquito. After that, extracting the feature takes place. The researchers used the help of Image Texture Analysis in extracting features.

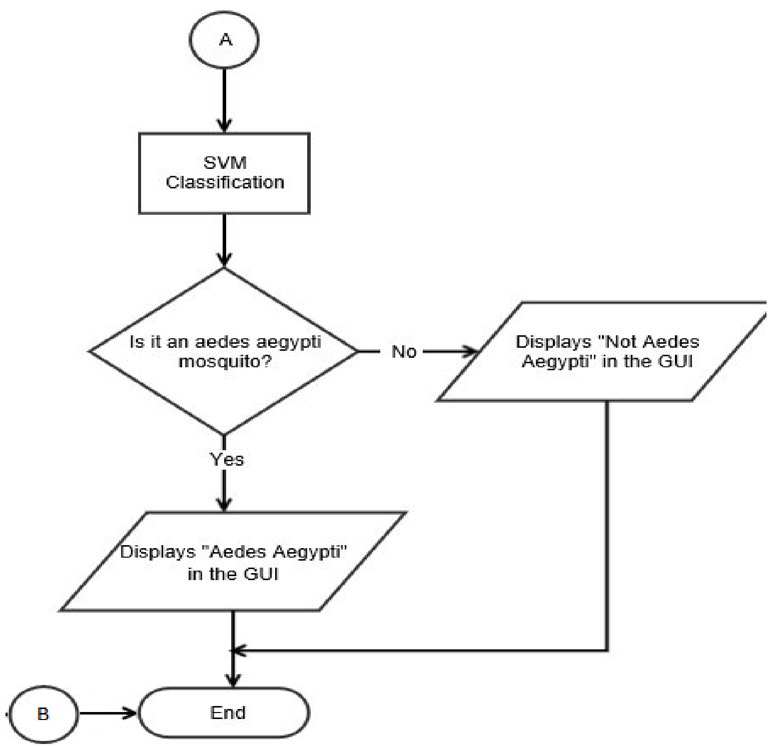


Figure 3.6.2 Continuation of System Flowchart

Figure 3.6.2 shows that once the features of the mosquito are extracted, the classification process by SVM takes place. The result of training the SVM with preprocessed image allows the system to classify whether the captured image was a matched from an Aedes Aegypti mosquito or not. Lastly, after the image captured of the mosquito was classified, a notification message appeared if it is an Aedes Aegypti mosquito. Otherwise, a message that says it was a not an Aedes Aegypti.

Hardware and Software Development

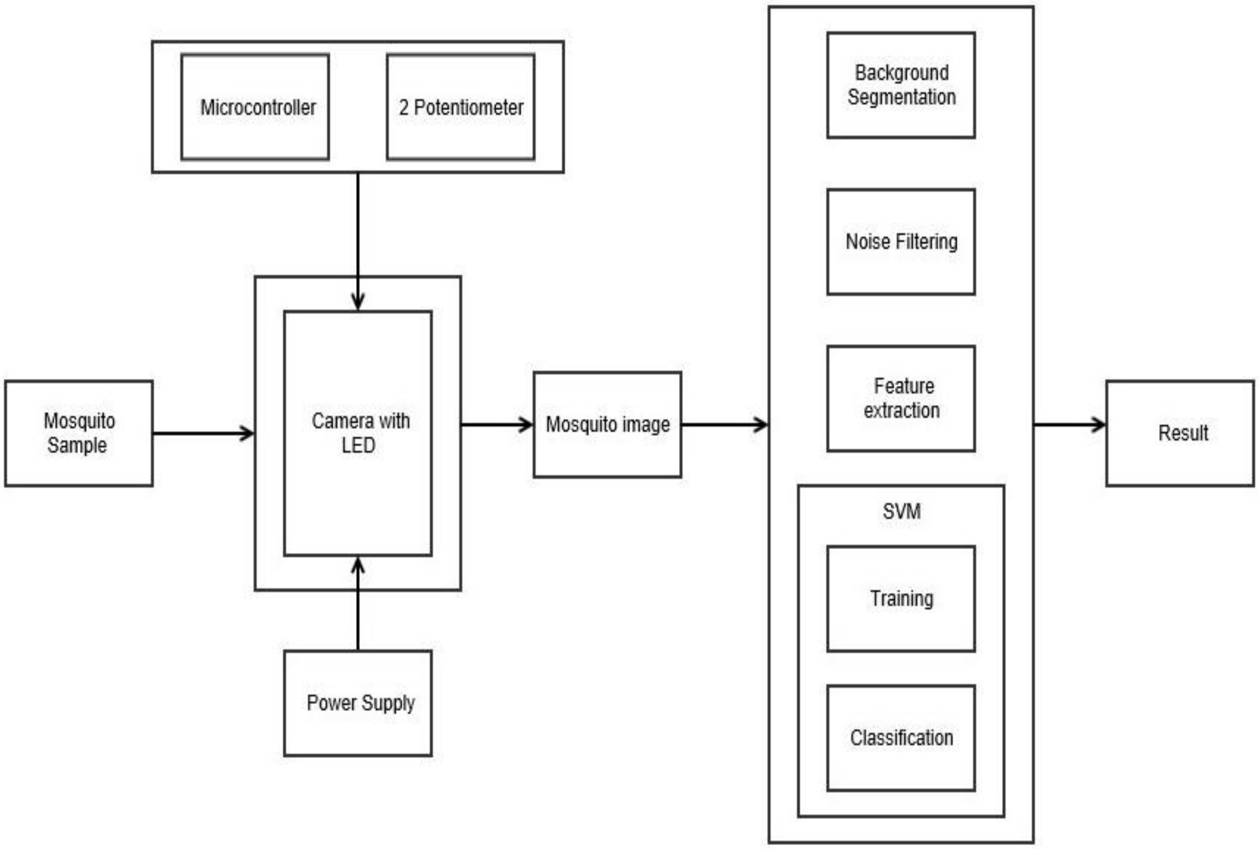


Figure 3.7 Hardware and Software Block Diagram

Figure 3.7 explains the implementation of the hardware and software development of the system.

Hardware Development

Hardware Requirements

The following components were used for the hardware development of the system.

1. Camera

Figure 3.8 Microscopic Camera

Image processing always involves an image acquisition. This is composed of a camera with 500x optical zoom or 5x digital zoom that was used to have a better result of the image that was captured. This specification is more likely for capturing insects. This camera was operated manually in order to take an image of your choice, since you can see in the system what you are capturing.

2. Power Supply

A power supply powers up the camera and has an output of 5V-DC.

3. Microcontroller

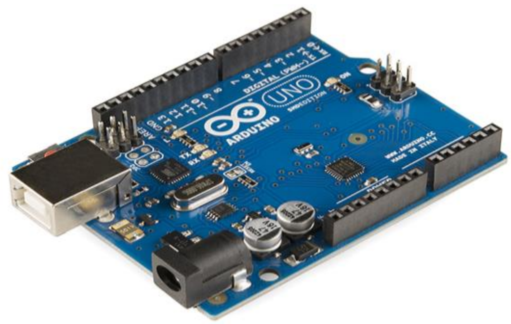


Figure 3.9 Arduino Uno

The microcontroller that was used in the system works between the camera and the computer system. This specific microcontroller is an Arduino Uno. Arduino Uno is a type of microcontroller board that is based on the ATmega328. It includes14 digital input/output pins, with 6 analog inputs, a 16 MHz ceramic resonator and a USB connection, with a power jack, and an ICSP header, and even a reset button. It is used for controlling the camera in terms of its rotation. The user controls the rotation of the camera through the programmed microcontroller. This microcontroller is programmed using the Integrated Development Environment (IDE) which is a multiplatform environment.

4. Potentiometer

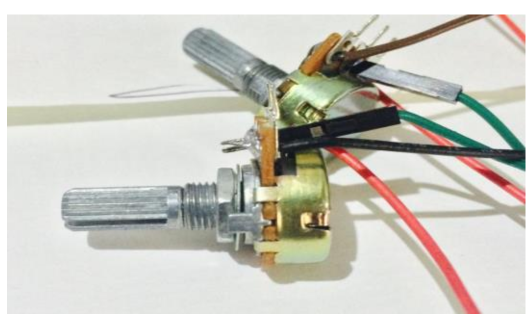


Figure 3.10 Two potentiometers

There are two potentiometers that controls the movement of the camera. It is used by the researcher to tilt and pan the view of the camera depending on which view the features of mosquito will be seen clearly. The two potentiometers were also connected to the Arduino Uno microcontroller.

5. Two servo motors

Figure 3.11 Servo Motors

These two motors were needed for the camera to be able to change its view either up and down or left and right. The two motors were setup to where the camera is installed. One motor is responsible for the tilting of the camera and one motor is for the panning.

Software Development

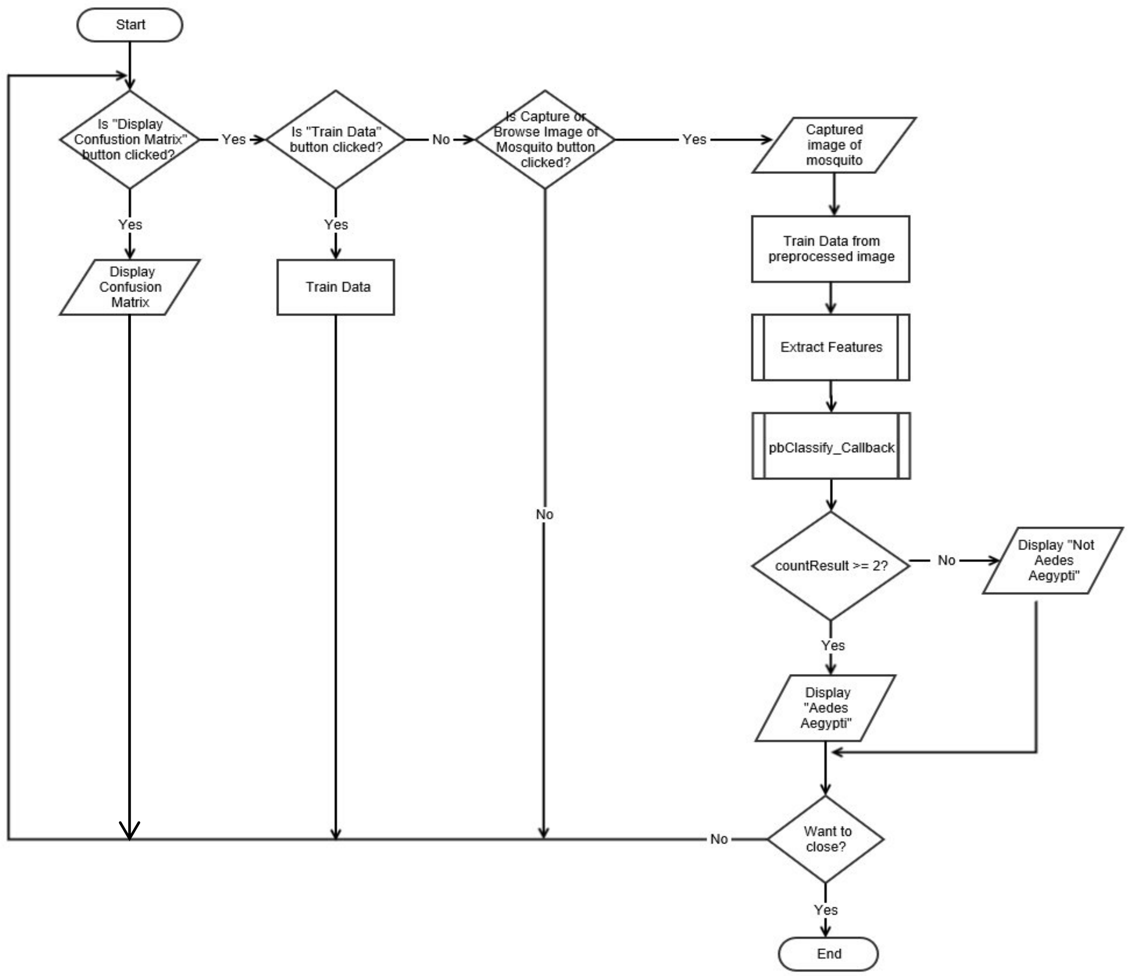


Figure 3.12 Software Flowchart

Figure 3.12 shows how the software of the system developed and works. In the graphical user interface, the system will display a button for confusion matrix. If the button was clicked, the system will display the confusion matrix. When the Train Data is clicked, the system will undergo training for identifying features of Aedes Aegypti. Now, when the Capture or Browse Image of Mosquito was clicked, the system lets the researcher capture image of mosquito, on the other hand, if it is not clicked, the system can be closed if the researcher wants to exit. After mosquito image was captured, it will undergo preprocessing of image and from that it was used for the training. Training undergoes through preprocessing that includes background segmentation and noise filtering. Once preprocessing is done, the image is trained in SVM. After being trained, the features of mosquito are extracted with the help of Image Texture Analysis, and then the values that have been produced are given to SVM Classification. If countResult >= 2, then the output is identified as Aedes Aegypti, if not it is identified as not Aedes Aegypti.

MATLAB

Matrix Laboratory or MATLAB application is used for developing an application that extracts specific features of a captured mosquito. MATLAB was used for making an algorithm that will match the image captured of a mosquito to the original image that is stored. For better, user friendly interface, the development of the GUI of the application was done using MATLAB.

Image processing

The process that takes place in image processing involves the background image segmentation, removing of noise in the image that is efficient for determining the pixel value. These digital image processing techniques were used for the preparation of image pattern recognition.

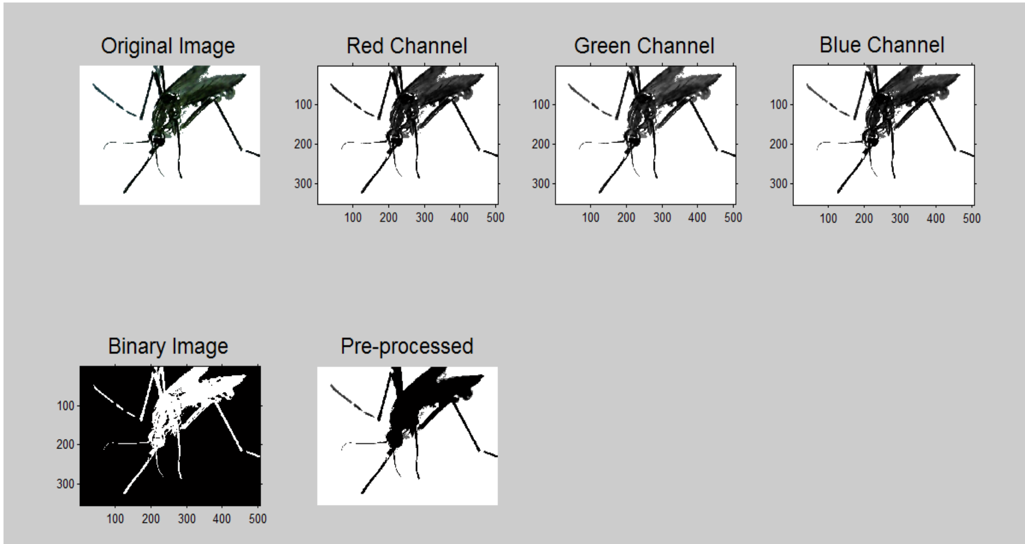
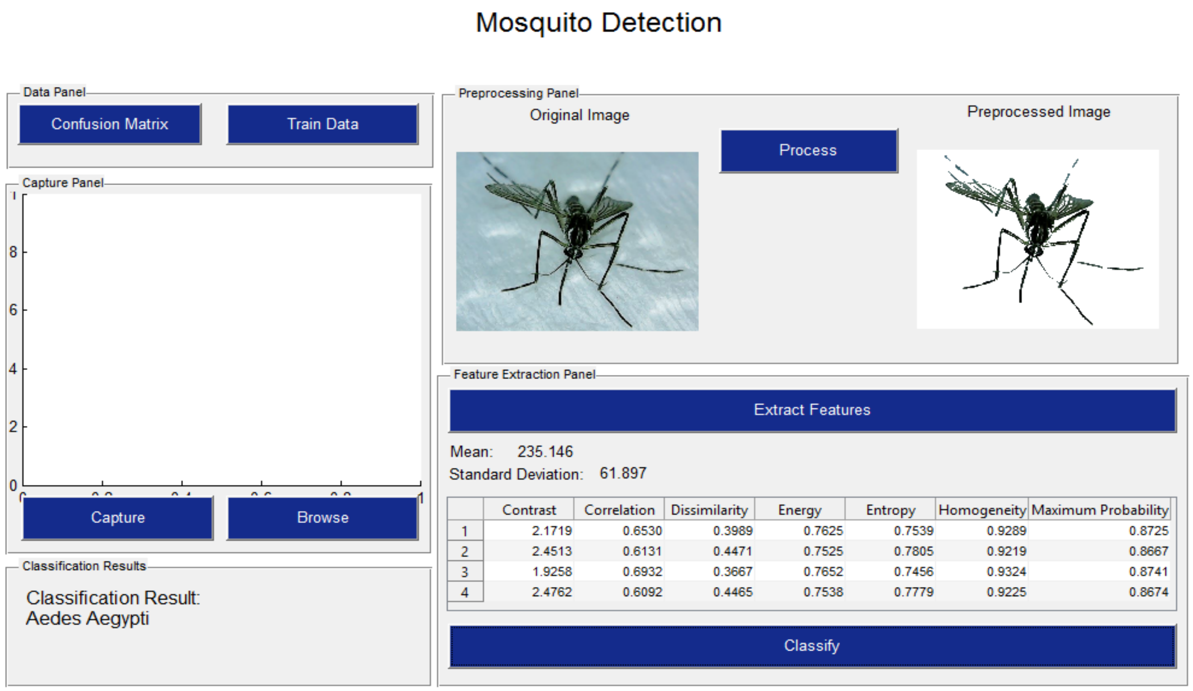


Figure 3.13 Preprocessed Image

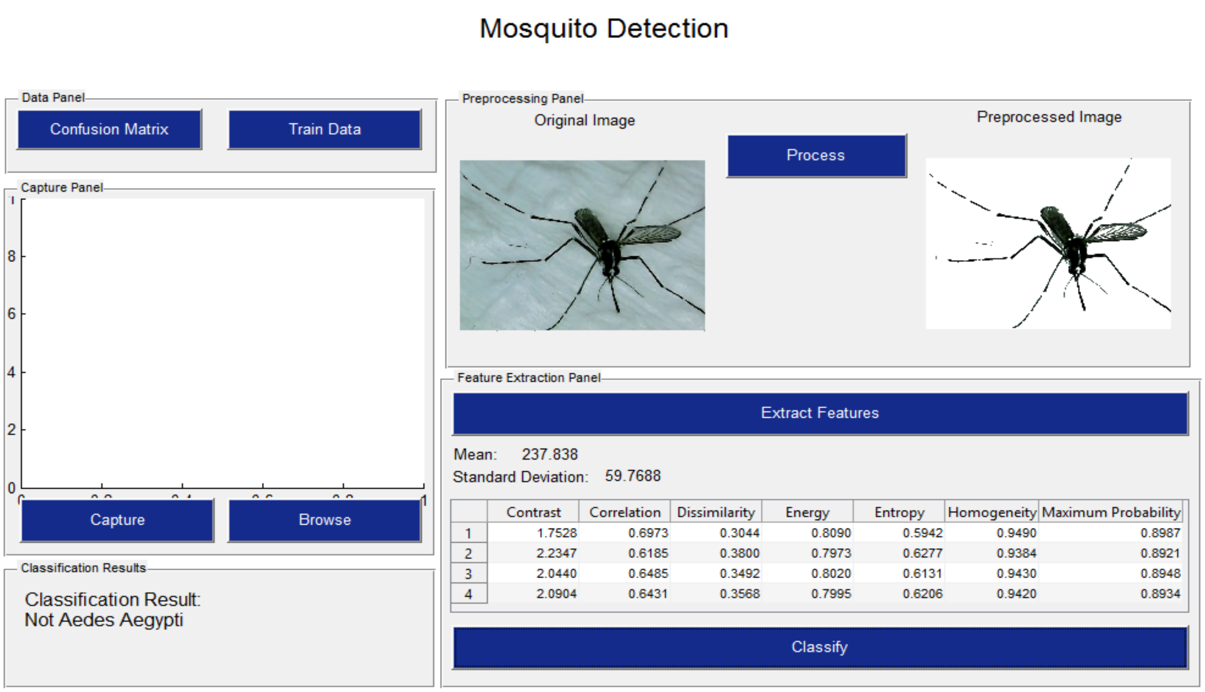
Figure 3.13 shows the output of an image that has been preprocessed by the system. From an original image, it was first processed by removing the red, green and blue channel or color of the image. Then, these images were combined to form its binary image to mask it out from the original image and form the shape of the mosquito. From getting its binary image, background segmentation and noise filtering was performed to get the preprocessed image.

Feature Extraction

Feature extraction was used for the separation of the needed data to the unnecessary, where through SVM or Support Vector Machine it classifies or indicates if the image captured of mosquito is a dengue carrier or not. The researcher used the help of SIFT and RANSAC to extract the unique features of mosquito, while Image Texture Analysis for the whole detection of mosquito.



(a)



(b)

Figure 3.14 Extraction and Classification, (a) Aedes Aegypti (b) Not Aedes Aegypti

This figure 3.14 shows the values that have been extracted from the images and how it was classified. The sample images were classified as Aedes Aegypti (a) and Not Aedes Aegypti (b). SVM classifies the images using the data from trained images in Appendix C.

Graphical Model of SVM

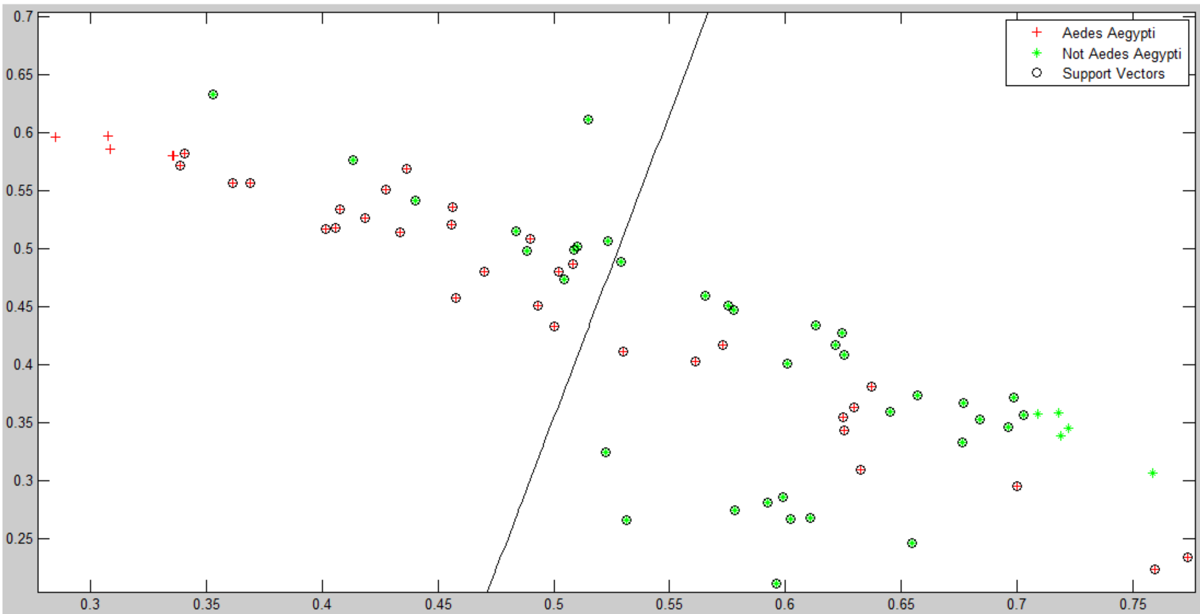


Figure 3.15 Graph of mean and standard deviation properties

Figure 3.15 shows the sample SVM model using the mean and standard deviation of trained images of an Aedes Aegypti mosquito. The x-axis represents the mean, while the y-axis represents the standard deviation. The line between the Aedes Aegypti and Not Aedes Aegypti is the separating hyperplane which decides what result comes out.

Results and Discussion

The following tables were used to show that the proposed system can detect a captured image of an Aedes Aegypti mosquito as an Aedes Aegypti from extracting its features and using SVM for classification.

Table 3.1 Characteristics of an Aedes Aegypti

	Head			Thorax			Abdomen
	Proboscis	Pulps	Clypeus	Wing scales	Leg color: Tarsal segment 5	Basal bands 2*6 for hind legs and3 for mid and fore legs	Scutum
Color	Dark	Tipped with silvery -white	White Scales	Dark	Entirely white	White	Lyre shaped, white scales

Table 3.1 shows the characteristics of Aedes Aegypti Mosquito that may be used to detect or to determine whether the captured image is an Aedes Aegypti or not. The characteristics are divided into three basing from where it is located on the body of the Aedes Mosquito. Based from what the researcher conducted, they used the only 5 features or characteristics of an Aedes Aegypti named: proboscis, pulps, clypeus, wing scales and scutum, because of its uniqueness to other mosquitoes. These features were also the most visible among the 7 features of the Aedes Aegypti mosquito.

Formula for the accuracy test:

$$\text{Accuracy} = \frac{subparts\ classified\ correctly}{total\ subparts}\ x\ 100\% \qquad (3.1)$$

Characteristics of Aedes Aegypti		Predicted Characteristics (model)				
Actual Characteristics (target)		Clypeus	Probocis	Pulps	Scutum	Wingscales
	Clypeus	3	0	0	0	0
	Probocis	0	3	0	0	0
	Pulps	0	0	3	0	0
	Scutum	0	0	0	3	0
	Wingscales	0	0	0	0	3

Table 3.2 Confusion Matrix for Characteristics of Aedes Aegypti

The researchers trained the system using 10 images for each characteristics of an Aedes Aegypti mosquito such as Clypeus, Probocis, Pulps, Scutum and Wingscales. For each characteristic, 3 images were used to test the correctness of the system. In table 3.1, it shows that the system was able to identify each characteristic correctly. For example, the 3 images of Clypeus were identified as Clypeus. This is also true for the 4 remaining characteristics of an Aedes Aegypti. Figure 3.16 and Figure 3.17 shows the images that have been used for training and testing.

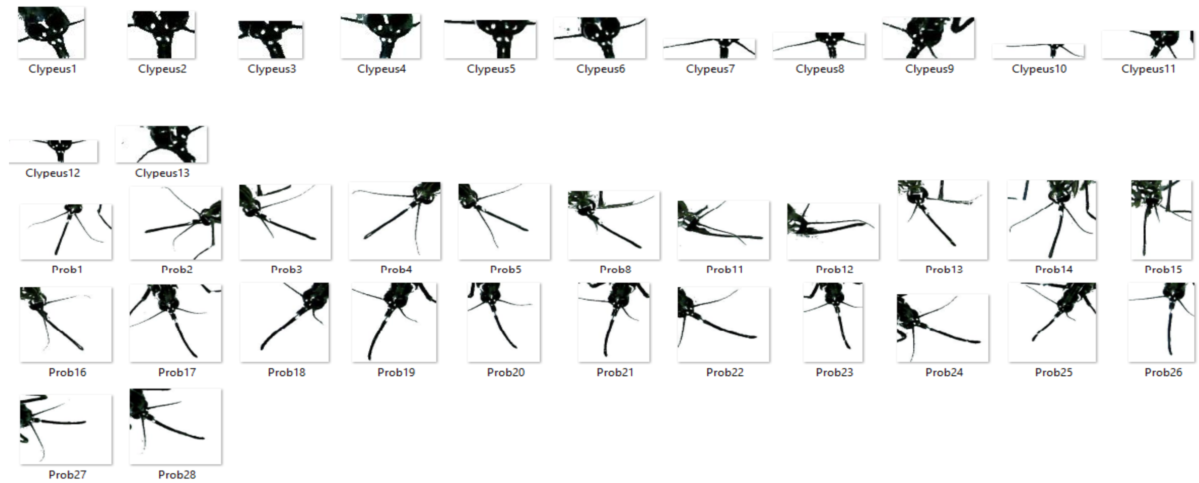


Figure 3.16 Images of Clypeus and Probocis

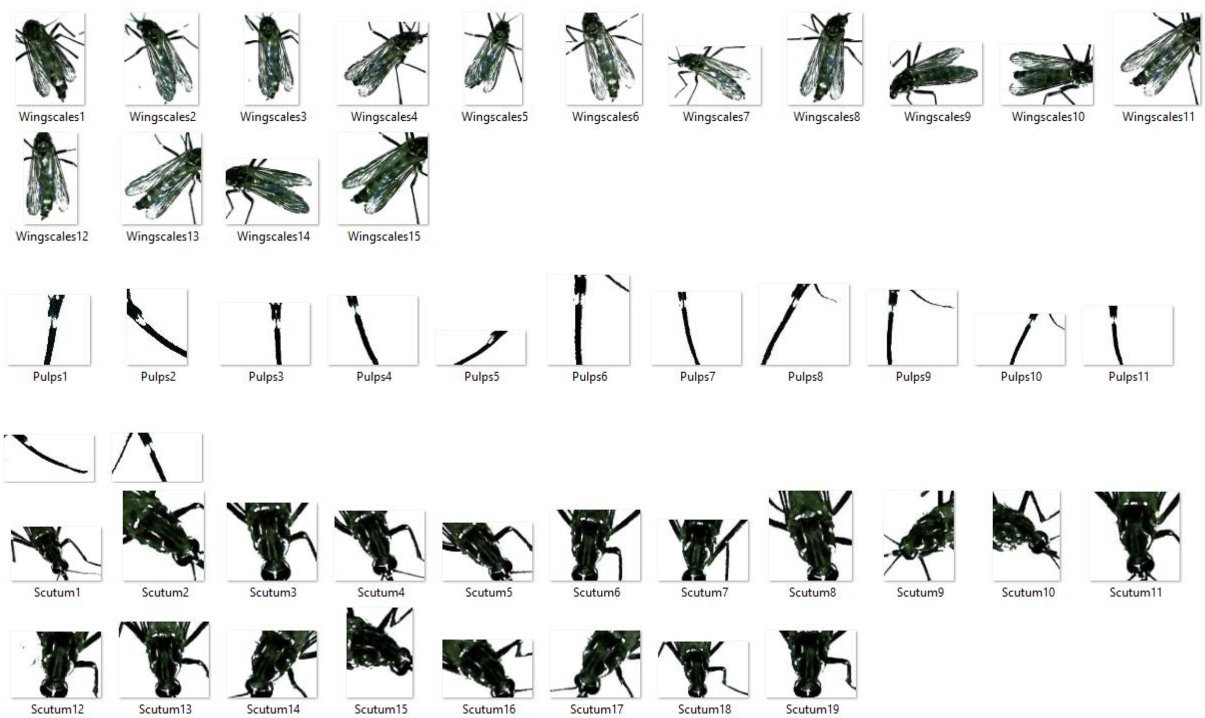


Figure 3.17 Images of Wingscales, Pulps and Scutum

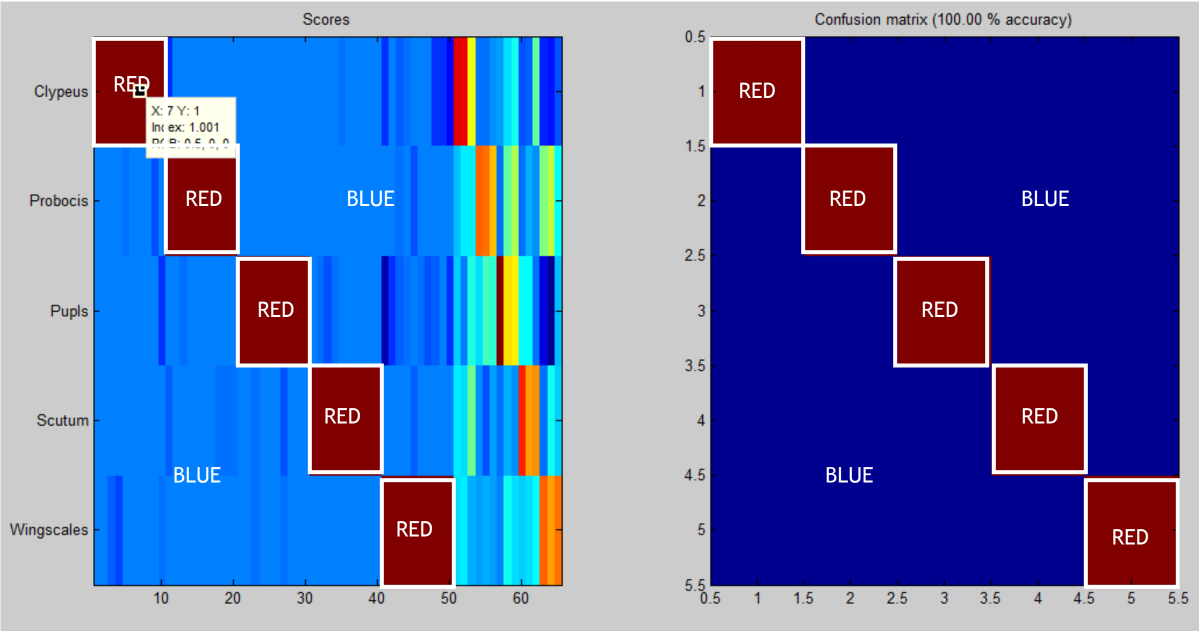


Figure 3.18 Image view of Confusion Matrix'- 1

This Figure 3.18 only shows the result of Confusion Matrix in training the SVM for the features of Aedes Aegypti, wherein the red color indicates that it is the characteristic of the Aedes Aegypti Mosquito. From the left part, this shows the 10 images that have been trained by the program, having an index 1 or nearly equal to 1; this indicates that the training that has been conducted was successful.

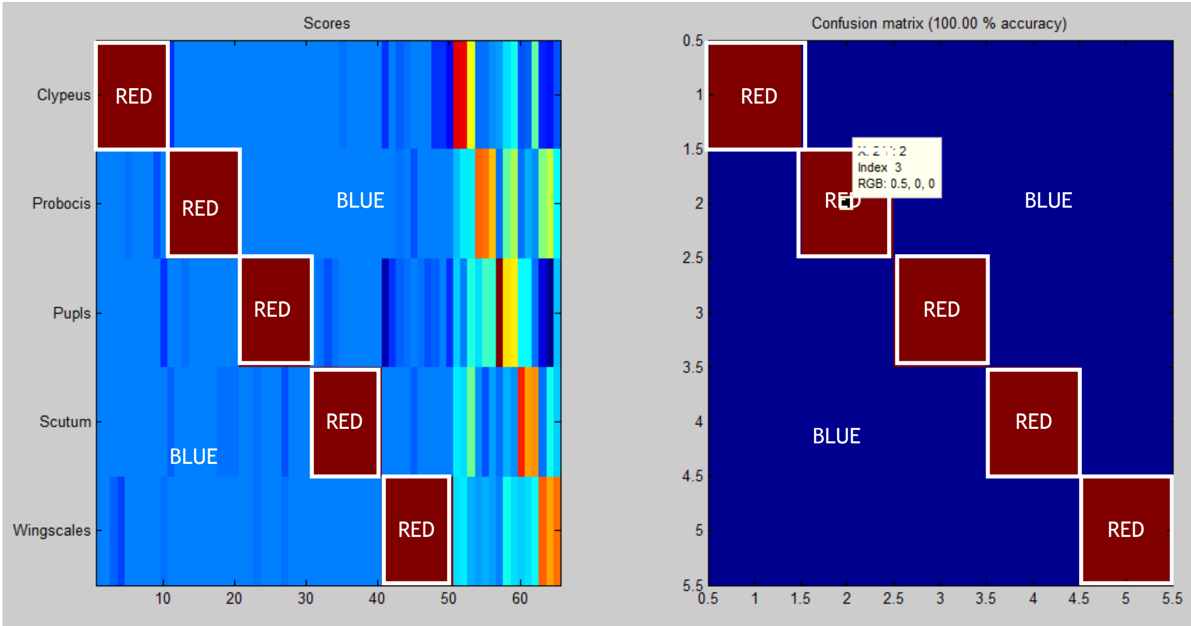


Figure 3.19 Image view of Confusion Matrix – 2

The researchers used 3 images in testing every characteristic of Aedes Aegypti and in order to indentify whether the test is correct, the value of index per feature must be equal to 3. From the right part of the figure 3.19, this shows that the results of the test were correct, since the value of the index for every characteristic is equal to 3, thus the detection of features was correct.

Table 3.3 Test 1 Detection of Aedes Aegypti in a container with samples of Aedes Aegypti and Not Aedes Aegypti

Captured Image	AedesAegypti (Yes or No)
1	No
2	No
3	No
4	Yes
5	No
6	Yes
7	No
8	No
9	No
10	No

From gathering 10 images of mosquito that was composed of 4 Aedes Aegypti and 6 Not Aedes Aegypti, table 3.3 shows the result of testing it randomly. Based from the test, 2 out of 4 images of Aedes Aegypti were detected and at the same time, 6 out of 6 images of Not Aedes Aegypti were also detected correctly by the system. This had happened due to some data that was not trained and/or to some parts of the image that was not supposed to be preprocessed.

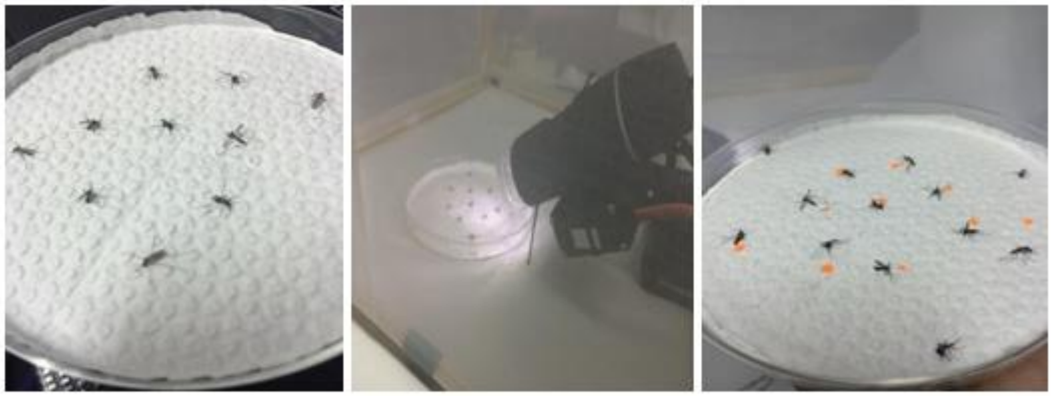


Figure 3.20 Test 1 Captured Images

Table 3.4 Test 2 Detection of Aedes Aegypti in a container with samples of Aedes Aegypti and Not Aedes Aegypti

Captured Image	AedesAegypti (Yes or No)
1	Yes
2	Yes
3	Yes
4	No
5	Yes
6	No
7	No
8	No

9	No
10	No

In order for the system to detect the mosquito correctly, the researchers conducted another random test for the images. This again was composed of 4 images of Aedes Aegypti mosquito and 6 images of Not Aedes Aegypti. Table 3.4 shows the result, where 4 out 4 images of Aedes Aegypti and 6 out of 6 images of Not Aedes Aegypti was correctly detected. Figure 3.21 below shows the images that were used for the test.

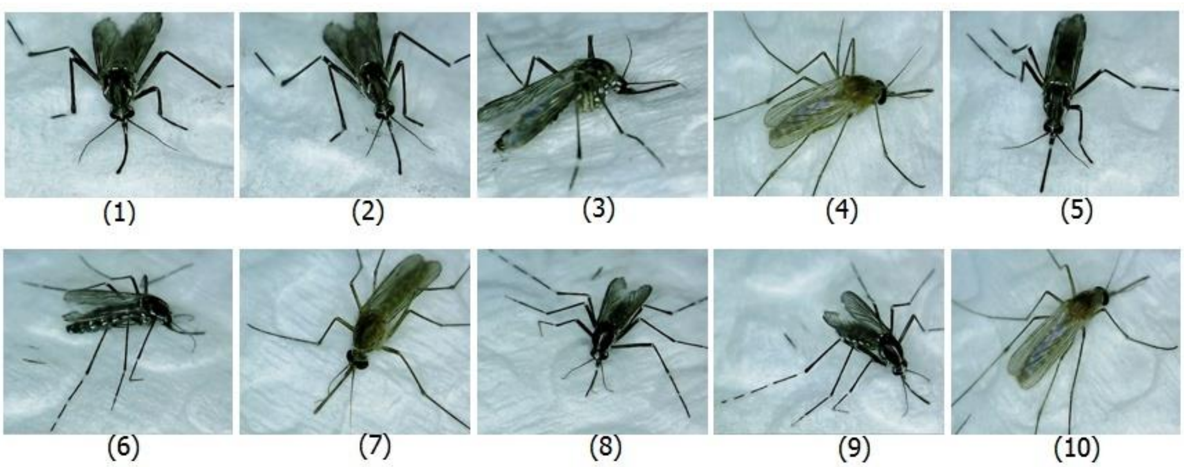


Figure 3.21 Test 2 Captured Images

		Predicted Mosquito	
Actual		**Aedes Aegypti**	**Not Aedes Aegypti**

Mosquito (target)	**Aedes Aegypti**	18	2
	Not Aedes Aegypti	1	19

Table 3.5 Testing for an Aedes Aegypti and Not Aedes Aegypti

Table 3.5 shows the result of testing 20 images of Aedes Aegypti mosquito and 20 not Aedes Aegypti mosquito. Based from the data that were gathered, 18 out of 20 images of Aedes Aegypti were detected correctly. Moreover, for the images of not Aedes Aegypti, 19 out of 20 images were detected correctly. Figure 3.22 below shows the images that have been tested by the system to get the percentage accuracy of classifying the mosquito correctly.

Figure 3.22 Forty Images for testing

Conclusion

From this paper, an accurate detection of a mosquito whether it is an Aedes Aegypti or not is presented, tested and implemented. The prototyped was designed for having a microscopic camera with a 500x optical zoom or 5x digital zoom, together with a proper implementation of the Arduino Uno microcontroller that was programmed to control the movement of the camera with the use of the potentiometers and servo motors. The researcher used the Support Vector Machine or SVM to classify the extracted features of the Aedes Aegypti Mosquito and gained a result of correctly classifying the 3 images per feature.

Also, the researcher used the image processing techniques such as the background segmentation, the noise filtering for preprocessing of the images. The researcher applied an Image Texture Analysis that also relies on SVM in order to match the captured image. These processes gave percentage accuracy from 80% and above were some discrepancies vary from the way the image is captured. Other reasons may also come from the preprocessing, where some details of the mosquito were also included from the segmentation process.

References

Rao, K.M.M. (2006*). Overview of Image Processing. Readings in Image Processing*, 1-5.

Young, I. T., J. J. Gerbrands and L. J. van Vliet (1995-2007). *Funadamentals of Image Processing*, 94.

Jahangir Alam S.M., H. Guoqing, and C. Chen (2014). *Characteristics Analysis and Detection Algorithm of Mosquitoes*. TELKOMNIKA Indonesian Journal of Electrical Engineering Volume 12 (7), 5368-5378.

Center for Disease Control and Prevention (CDC)(Jan3,2012). “*aegyptifactsheet*”. National Center for Emerging and Zoonotic Infectious Diseases.

ThulasiPriya C., K.Praveen, and A.Srividya (2013). *Monitoring Of Pest Insect Traps Using Image Sensors &Dspic*. International Journal Of Engineering Trends And Technology, Volume 4 (9), 4092.

Mundada R. G. (2013). *Detection and Classification of Pests in Greenhouse Using Image* Processing. *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE),* Volume **5** (6), 57-63.

Marques, O. (2011). *Practical Image and Video Processing Using MATLAB*, John Wiley & Sons, Inc., Hoboken, New Jersey.

Gonzales, R. C., R. E. Woods, and S. L. Eddins (2009). *Digital Image Processing Using MATLAB*, Second Edition, Gatesmark Publishing, United States of America.

Delos Angeles, Bethsedea Joy S., Dizon, Neil John A., Mardo, Mark Kevin N., *SupportVector Machine and Mel Frequency Cepstral Coefficient based Algorithm for Hand Gestures and Bidirectional Speech to Text Device*, MIT-October 2014

Pengrui Qiu,Ying Liang and Hui Rong (2013). *Image Mosaics Algorithm Based on SIFT Feature Point Matching andTransformation Parameters Automatically Recognizing*. Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)

Ruwen Schnabel Roland Wahl Reinhard Klein† (2007). *Efficient RANSAC for Point-Cloud Shape Detection*. Volume 0 (1981), Number 0 pp. 1–12

Ruth Bergman, Hila Nachlieli, Gitit Ruckenstein (March 2007). *Detection of Textured Areas in Images Using a Disorganization Indicator Based on Component Counts*. *HP Laboratories*. Israel HPL-2005-175(R.1)

Leen-Kiat Soh, Costas Tsatsoulis (1991). *Texture Analysis of SAR Sea Ice Imagery Using Gray Level Co-Occurrence Matrices*. University of Nebraska – Lincoln DigitalCommons@University of Nebraska – Lincoln. CSE Journal Articles

Andrew Ng. *CS229 Machine Learning Course Materials.* Standford University

Chapter 4

CONCLUSION

From this paper, an accurate detection of a mosquito whether it is an Aedes Aegypti or not is presented, tested and implemented. The prototyped was designed for having a microscopic camera with a 500x optical zoom or 5x digital zoom, together with a proper implementation of the Arduino Uno microcontroller that was programmed to control the movement of the camera with the use of the potentiometers and servo motors. The researcher used the Support Vector Machine or SVM to classify the extracted features of the Aedes Aegypti Mosquito and gained a result of correctly classifying the 3 images per feature.

Also, the researcher used the image processing techniques such as the background segmentation, the noise filtering for preprocessing of the images. The researcher applied an Image Texture Analysis that also relies on SVM in order to match the captured image. These processes gave percentage accuracy from 80% and above were some discrepancies vary from the way the image is captured. Other reasons may also come from the preprocessing, where some details of the mosquito were also included from the segmentation process.

Chapter 5

RECOMMENDATION

Based from the findings and results of the trainings and testings that the researchers had done, the following are recommended for the improvement of the system: The implementation of “moving state” of mosquito and its detection whether it is an Aedes Aegypti or not by implementing sensors for insects in the system. Another is a higher magnification of the microscopic camera for a better image quality whenever the mosquitos are captured since the system is still on manual focus. Also, the system still needs a faster approach in extracting the features of mosquito by implementing other techniques in Matlab, since the system took a long time in processing an image.

REFERENCES

Rao, K.M.M. (2006*). Overview of Image Processing. Readings in Image Processing*, 1-5.

Young, I. T., J. J. Gerbrands and L. J. van Vliet (1995-2007). *Funadamentals of Image Processing*, 94.

Jahangir Alam S.M., H. Guoqing, and C. Chen (2014). *Characteristics Analysis and Detection Algorithm of Mosquitoes*. TELKOMNIKA Indonesian Journal of Electrical Engineering Volume 12 (7), 5368-5378.

Center for Disease Control and Prevention (CDC)(Jan3,2012). "*aegyptifactsheet*". National Center for Emerging and Zoonotic Infectious Diseases.

Thulasi Priya C., K.Praveen, and A.Srividya (2013). *Monitoring Of Pest Insect Traps Using Image Sensors &Dspic*. International Journal Of Engineering Trends And Technology, Volume 4 (9), 4092.

Mundada R. G. (2013). *Detection and Classification of Pests in Greenhouse Using Image* Processing. *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*, Volume **5** (6), 57-63.

Marques, O. (2011). *Practical Image and Video Processing Using MATLAB*, John Wiley & Sons, Inc., Hoboken, New Jersey.

Gonzales, R. C., R. E. Woods, and S. L. Eddins (2009). *Digital Image Processing Using MATLAB*, Second Edition, Gatesmark Publishing, United States of America.

Delos Angeles, Bethsedea Joy S., Dizon, Neil John A., Mardo, Mark Kevin N., *SupportVector Machine and Mel Frequency Cepstral Coefficient based Algorithm for Hand Gestures and Bidirectional Speech to Text Device*, MIT-October 2014

Pengrui Qiu,Ying Liang and Hui Rong (2013). *Image Mosaics Algorithm Based on SIFT Feature Point Matching andTransformation Parameters Automatically Recognizing*. Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)

Ruwen Schnabel Roland Wahl Reinhard Klein† (2007). *Efficient RANSAC for Point-Cloud Shape Detection*. Volume 0 (1981), Number 0 pp. 1–12

Ruth Bergman, Hila Nachlieli, Gitit Ruckenstein (March 2007). *Detection of Textured Areas in Images Using a Disorganization Indicator Based on Component Counts. HP Laboratories*. Israel HPL-2005-175(R.1)

Leen-Kiat Soh, Costas Tsatsoulis (1991). *Texture Analysis of SAR Sea Ice Imagery Using Gray Level Co-Occurrence Matrices*. University of Nebraska – Lincoln DigitalCommons@University of Nebraska – Lincoln. CSE Journal Articles

APPENDICES

APPENDIX A

Matlab Codes

GUI

```
function varargout =
MosquitoGUI(varargin)
% MOSQUITOGUI MATLAB code for
MosquitoGUI.fig
%      MOSQUITOGUI, by itself, creates a
new MOSQUITOGUI or raises the
existing
%      singleton*.
%
%      H = MOSQUITOGUI returns the
handle to a new MOSQUITOGUI or the
handle to
%      the existing singleton*.
%
%
MOSQUITOGUI('CALLBACK',hObject,e
ventData,handles,...) calls the local
%      function named CALLBACK in
MOSQUITOGUI.M with the given input
arguments.
%
%
MOSQUITOGUI('Property','Value',...)
creates a new MOSQUITOGUI or raises
the
%      existing singleton*.  Starting from
the left, property value pairs are
%      applied to the GUI before
MosquitoGUI_OpeningFcn gets called.
An
%      unrecognized property name or
invalid value makes property application
%      stop.  All inputs are passed to
MosquitoGUI_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's
Tools menu.  Choose "GUI allows only
one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA,
GUIHANDLES

% Edit the above text to modify the
response to help MosquitoGUI

% Last Modified by GUIDE v2.5 03-Oct-
2015 23:33:17

% Begin initialization code - DO NOT
EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',
mfilename, ...
                   'gui_Singleton',
gui_Singleton, ...
                   'gui_OpeningFcn',
@MosquitoGUI_OpeningFcn, ...
                   'gui_OutputFcn',
@MosquitoGUI_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
   gui_State.gui_Callback =
str2func(varargin{1});
end

if nargout
   [varargout{1:nargout}] =
gui_mainfcn(gui_State, varargin{:});
else
   gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

function DisableAllButton(hObject,
handles)
set(handles.pbConfusionMatrix, 'Enable',
'off');
set(handles.pbTrainData, 'Enable', 'off');
```

```
set(handles.pbCapture, 'Enable', 'off');

set(handles.pbProcess, 'Enable', 'off');
set(handles.pbBrowse, 'Enable', 'off');
set(handles.pbFeatureExtraction, 'Enable',
'off');

set(handles.pbClassify, 'Enable', 'off');



function EnableAllButton(hObject,
handles)
set(handles.pbConfusionMatrix, 'Enable',
'on');
set(handles.pbTrainData, 'Enable', 'on');
set(handles.pbCapture, 'Enable', 'on');

set(handles.pbProcess, 'Enable', 'on');
set(handles.pbBrowse, 'Enable', 'on');
set(handles.pbFeatureExtraction, 'Enable',
'on');

set(handles.pbClassify, 'Enable', 'on')

% --- Executes just before MosquitoGUI is
made visible.
function
MosquitoGUI_OpeningFcn(hObject,
eventdata, handles, varargin)

set(handles.axisOriginalImage, 'Visible',
'off');
set(handles.axisPreprocessedImage,
'Visible', 'off');
set(handles.pbProcess, 'Visible', 'off');

set(handles.pbFeatureExtraction, 'Visible',
'off');
set(handles.lblMeanText, 'Visible', 'off');
set(handles.lblMeanValue, 'Visible', 'off');
set(handles.lblStandardDeviationText,
'Visible', 'off');
set(handles.lblStandardDeviationValue,
'Visible', 'off');
set(handles.tableFeatures, 'Visible', 'off');
set(handles.pbClassify, 'Visible', 'off');

set(handles.lblClassificationResult,'Visible
', 'off');

handles.output = hObject;
axes(handles.axisCameraCapture);
vid = videoinput('winvideo', 2,
'YUY2_640x480');
src = getselectedsource(vid);

handles.hImage = image(zeros(480,640,
3), 'Parent',handles.axisCameraCapture);
preview(vid,handles.hImage);
handles.imageCaptured = 'none';
handles.imagePreprocessed = 'none';
handles.featureExtractionResult = 'none';
% Update handles structure
guidata(hObject, handles);

% UIWAIT makes MosquitoGUI wait for
user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are
returned to the command line.
function varargout =
MosquitoGUI_OutputFcn(hObject,
eventdata, handles)
% varargout  cell array for returning
output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a
future version of MATLAB
% handles    structure with handles and
user data (see GUIDATA)

% Get default command line output from
handles structure
varargout{1} = handles.output;


% --- Executes on button press in
pbConfusionMatrix.
```

```
function
pbConfusionMatrix_Callback(hObject,
eventdata, handles)
clc;
DisableAllButton(hObject, handles);
ConfusionMatrix;
EnableAllButton(hObject, handles);
display('Showing Confusion Matrix...');

% --- Executes on button press in
pbTrainData.
function pbTrainData_Callback(hObject,
eventdata, handles)
clc;
DisableAllButton(hObject, handles);
TrainData;
EnableAllButton(hObject, handles)
display('Done Training Data...');

% --- Executes on button press in
pbCapture.
function pbCapture_Callback(hObject,
eventdata, handles)
set(handles.axisOriginalImage, 'Visible',
'off');
set(handles.axisPreprocessedImage,
'Visible', 'off');
set(handles.pbProcess, 'Visible', 'off');

set(handles.pbFeatureExtraction, 'Visible',
'off');
set(handles.lblMeanText, 'Visible', 'off');
set(handles.lblMeanValue, 'Visible', 'off');
set(handles.lblStandardDeviationText,
'Visible', 'off');
set(handles.lblStandardDeviationValue,
'Visible', 'off');
set(handles.tableFeatures, 'Visible', 'off');
set(handles.pbClassify, 'Visible', 'off');
set(handles.lblClassificationResult,'Visible
', 'off');
DisableAllButton(hObject, handles);

axes(handles.axisOriginalImage);
handles.imageCaptured =
getimage(handles.axisCameraCapture);
handles.imagePreprocessed = 'none';
imshow(handles.imageCaptured);

cla(handles.axisPreprocessedImage);
set(handles.pbProcess, 'Visible', 'on');

EnableAllButton(hObject, handles);

guidata(hObject, handles);

% --- Executes on button press in
pbBrowse.
function pbBrowse_Callback(hObject,
eventdata, handles)
set(handles.axisOriginalImage, 'Visible',
'off');
set(handles.axisPreprocessedImage,
'Visible', 'off');
set(handles.pbProcess, 'Visible', 'off');

set(handles.pbFeatureExtraction, 'Visible',
'off');
set(handles.lblMeanText, 'Visible', 'off');
set(handles.lblMeanValue, 'Visible', 'off');
set(handles.lblStandardDeviationText,
'Visible', 'off');
set(handles.lblStandardDeviationValue,
'Visible', 'off');
set(handles.tableFeatures, 'Visible', 'off');
set(handles.pbClassify, 'Visible', 'off');
set(handles.lblClassificationResult,'Visible
', 'off');

DisableAllButton(hObject, handles);

[filename, pathname] =
uigetfile({'*.jpg;*.png;*.bmp;', 'Image
Files (*.jpg,*.png,*.bmp)';}, 'Select
mosquito image sample');

if(pathname ~= 0)
   axes(handles.axisOriginalImage);
   handles.imageCaptured =
imread(strcat(pathname,filename));
   handles.imagePreprocessed = 'none';
   imshow(handles.imageCaptured);
```

```
    cla(handles.axisPreprocessedImage);
    set(handles.pbProcess, 'Visible', 'on');
end

EnableAllButton(hObject, handles);

guidata(hObject, handles);

% --- Executes on button press in
pbProcess.
function pbProcess_Callback(hObject,
eventdata, handles)
set(handles.lblMeanText, 'Visible', 'off');
set(handles.lblMeanValue, 'Visible', 'off');
set(handles.lblStandardDeviationText,
'Visible', 'off');
set(handles.lblStandardDeviationValue,
'Visible', 'off');
set(handles.tableFeatures, 'Visible', 'off');
set(handles.pbClassify, 'Visible', 'off');
set(handles.lblClassificationResult,'Visible
', 'off');

DisableAllButton(hObject, handles);
pause(0.1);

imageFileResult =
Preprocessing(handles.imageCaptured);

axes(handles.axisPreprocessedImage);
handles.imagePreprocessed =
imageFileResult;
handles.featureExtractionResult = 'none';
imshow(imageFileResult);

EnableAllButton(hObject, handles);

set(handles.pbFeatureExtraction, 'Visible',
'on');

guidata(hObject, handles);

% --- Executes on button press in
pbFeatureExtraction.
function
pbFeatureExtraction_Callback(hObject,
eventdata, handles)

if(~strcmp(handles.imagePreprocessed,'no
ne'))
    DisableAllButton(hObject, handles);
    pause(0.1);
    results =
TextureFeatureExtraction(handles.imagePr
eprocessed);
    handles.featureExtractionResult =
results;
    EnableAllButton(hObject, handles);

    set(handles.tableFeatures, 'Data',
results(1:4,3:9));
    set(handles.lblMeanValue, 'String',
num2str(results(1,1)));
    set(handles.lblStandardDeviationValue,
'String', num2str(results(1,2)));
    set(handles.lblMeanText, 'Visible', 'on');
    set(handles.lblMeanValue, 'Visible',
'on');
    set(handles.lblStandardDeviationText,
'Visible', 'on');
    set(handles.lblStandardDeviationValue,
'Visible', 'on');
    set(handles.tableFeatures, 'Visible',
'on');
    set(handles.pbClassify, 'Visible', 'on');

set(handles.lblClassificationResult,'Visible
', 'off');
    guidata(hObject, handles);
else

end


% --- Executes on button press in
pbClassify.
function pbClassify_Callback(hObject,
eventdata, handles)
```

```
set(handles.lblClassificationResult,'Visible
', 'on');
if(~strcmp(handles.featureExtractionResul
t,'none'))
  load('Data/svmStructure.mat');
  load('Data/csMean.mat');
  load('Data/csStandardDeviation.mat');
  load('Data/csContrast.mat');

  result = handles.featureExtractionResult;

  result(:,1) = mapminmax('apply',
result(:,1), csMean);
  result(:,2) = mapminmax('apply',
result(:,2), csStandardDeviation);
  result(:,3) = mapminmax('apply',
result(:,3), csContrast);

  species =
svmclassify(svmStructure,result);

  %count Aedes Aegypti
  countResult =
sum(strcmp(species(:),'Aedes Aegypti'));
  if(countResult >= 2)

set(handles.lblClassificationResult,'String',
{'Classification Result:'; 'Aedes
Aegypti';});
  else

set(handles.lblClassificationResult,'String',
{'Classification Result:'; 'Not Aedes
Aegypti';});
  end
end
```

Confusion Matrix

```
function ConfusionMatrix()
configuration.calDir = 'Images/Features' ;
configuration.dataDir = 'Data/' ;
configuration.autoDownloadData = true ;
configuration.numTrain = 10;
configuration.numTest = 3;
configuration.numClasses = 102 ;
configuration.numWords = 600 ;
configuration.numSpatialX = [2 4] ;
configuration.numSpatialY = [2 4] ;
configuration.quantizer = 'kdtree' ;
configuration.svm.C = 10 ;

configuration.svm.solver = 'sdca' ;

configuration.svm.biasMultiplier = 1 ;
configuration.phowOpts = {'Step', 3} ;
configuration.clobber = false ;
configuration.tinyProblem = true ;
configuration.prefix = 'baseline' ;
configuration.randSeed = 1 ;

if configuration.tinyProblem
  configuration.prefix = 'tiny' ;
  configuration.numClasses = 5 ;
  configuration.numSpatialX = 3 ;
  configuration.numSpatialY = 3 ;
  configuration.numWords = 300 ;
  configuration.phowOpts = {'Verbose', 2,
'Sizes', 7, 'Step', 5} ;
end

configuration.vocabPath =
fullfile(configuration.dataDir,
[configuration.prefix '-vocab.mat']) ;
configuration.histPath =
fullfile(configuration.dataDir,
[configuration.prefix '-hists.mat']) ;
configuration.modelPath =
fullfile(configuration.dataDir,
[configuration.prefix '-model.mat']) ;
configuration.resultPath =
fullfile(configuration.dataDir,
[configuration.prefix '-result']) ;

randn('state',configuration.randSeed) ;
rand('state',configuration.randSeed) ;
vl_twister('state',configuration.randSeed) ;

% ---------------------------------------------
-------------------
```

```
%
Setup data
% -----------------------------------------------
--------------------
classes = dir(configuration.calDir) ;
classes = classes([classes.isdir]) ;
classes =
{classes(3:configuration.numClasses+2).n
ame} ;

images = {} ;
imageClass = {} ;
for ci = 1:length(classes)
  ims = dir(fullfile(configuration.calDir,
classes{ci}, '*.jpg'))' ;
  ims = vl_colsubset(ims,
configuration.numTrain +
configuration.numTest) ;
  ims =
cellfun(@(x)fullfile(classes{ci},x),{ims.na
me},'UniformOutput',false) ;
  images = {images{:}, ims{:}} ;
  imageClass{end+1} = ci *
ones(1,length(ims)) ;
end
selTrain = find(mod(0:length(images)-1,
configuration.numTrain+configuration.nu
mTest) < configuration.numTrain) ;
selTest = setdiff(1:length(images),
selTrain) ;
imageClass = cat(2, imageClass{:}) ;

model.classes = classes ;
model.phowOpts =
configuration.phowOpts ;
model.numSpatialX =
configuration.numSpatialX ;
model.numSpatialY =
configuration.numSpatialY ;
model.quantizer = configuration.quantizer
;
model.vocab = [] ;
model.w = [] ;
model.b = [] ;
model.classify = @classify ;
```

```
% -----------------------------------------------
--------------------
%                                    Train
vocabulary
% -----------------------------------------------
--------------------

if ~exist(configuration.vocabPath) ||
configuration.clobber

  % Get some PHOW descriptors to train
the dictionary
  selTrainFeats = vl_colsubset(selTrain,
30) ;
  descrs = {} ;
  %for ii = 1:length(selTrainFeats)
  parfor ii = 1:length(selTrainFeats)
    im =
imread(fullfile(configuration.calDir,
images{selTrainFeats(ii)})) ;
    im = standarizeImage(im) ;
    [drop, descrs{ii}] = vl_phow(im,
model.phowOpts{:}) ;
  end

  descrs = vl_colsubset(cat(2, descrs{:}),
10e4) ;
  descrs = single(descrs) ;

  % Quantize the descriptors to get the
visual words
  vocab = vl_kmeans(descrs,
configuration.numWords, 'verbose',
'algorithm', 'elkan', 'MaxNumIterations',
50) ;
  save(configuration.vocabPath, 'vocab') ;
else
  load(configuration.vocabPath) ;
end

model.vocab = vocab ;

if strcmp(model.quantizer, 'kdtree')
  model.kdtree = vl_kdtreebuild(vocab) ;
end
```

```
% -----------------------------------------------
--------------------
%                              Compute
spatial histograms
% -----------------------------------------------
--------------------

if ~exist(configuration.histPath) ||
configuration.clobber
  hists = {} ;
  parfor ii = 1:length(images)
  % for ii = 1:length(images)
    fprintf('Processing %s (%.2f %%)\n',
images{ii}, 100 * ii / length(images)) ;
    im =
imread(fullfile(configuration.calDir,
images{ii})) ;
    hists{ii} = getImageDescriptor(model,
im);
  end

  hists = cat(2, hists{:}) ;
  save(configuration.histPath, 'hists') ;
else
  load(configuration.histPath) ;
end

% -----------------------------------------------
--------------------
%                                   Compute
feature map
% -----------------------------------------------
--------------------

psix = vl_homkermap(hists, 1, 'kchi2',
'gamma', .5) ;

% -----------------------------------------------
--------------------
%
Train SVM
% -----------------------------------------------
--------------------

if ~exist(configuration.modelPath) ||
configuration.clobber
  switch configuration.svm.solver
    case {'sgd', 'sdca'}
      lambda = 1 / (configuration.svm.C *
length(selTrain)) ;
      w = [] ;
      parfor ci = 1:length(classes)
        perm = randperm(length(selTrain)) ;
        fprintf('Training model for class
%s\n', classes{ci}) ;
        y = 2 * (imageClass(selTrain) == ci) -
1 ;
        [w(:,ci) b(ci) info] =
vl_svmtrain(psix(:, selTrain(perm)),
y(perm), lambda, ...
          'Solver', configuration.svm.solver, ...
          'MaxNumIterations', 50/lambda, ...
          'BiasMultiplier',
configuration.svm.biasMultiplier, ...
          'Epsilon', 1e-3);
      end

    case 'liblinear'
      svm = train(imageClass(selTrain)', ...

sparse(double(psix(:,selTrain))),  ...
                  sprintf(' -s 3 -B %f -c %f', ...

configuration.svm.biasMultiplier,
configuration.svm.C), ...
                  'col') ;
      w = svm.w(:,1:end-1)' ;
      b =  svm.w(:,end)' ;
  end

  model.b =
configuration.svm.biasMultiplier * b ;
  model.w = w ;

  save(configuration.modelPath, 'model') ;
else
  load(configuration.modelPath) ;
end

% -----------------------------------------------
--------------------
```

```
%                                    Test SVM
and evaluate
% ----------------------------------------------
-------------------

% Estimate the class of the test images
scores = model.w' * psix + model.b' *
ones(1,size(psix,2)) ;
[drop, imageEstClass] = max(scores, [], 1)
;

% Compute the confusion matrix
idx = sub2ind([length(classes),
length(classes)], ...
          imageClass(selTest),
imageEstClass(selTest)) ;
confus = zeros(length(classes)) ;
confus = vl_binsum(confus,
ones(size(idx)), idx) ;

% Plots
figure(1) ; clf;
subplot(1,2,1) ;
imagesc(scores(:,[selTrain selTest])) ;
title('Scores') ;
set(gca, 'ytick', 1:length(classes),
'yticklabel', classes) ;
subplot(1,2,2) ;
imagesc(confus) ;
title(sprintf('Confusion matrix (%.2f %%
accuracy)', ...
          100 *
mean(diag(confus)/configuration.numTest
) )) ;
print('-depsc2', [configuration.resultPath
'.ps']) ;
save([configuration.resultPath '.mat'],
'confus', 'configuration') ;

% ----------------------------------------------
------------------------
function im = standarizeImage(im)
% ----------------------------------------------
------------------------

im = im2single(im) ;
if size(im,1) > 480, im = imresize(im, [480
NaN]) ; end

% ----------------------------------------------
------------------------
function hist = getImageDescriptor(model,
im)
% ----------------------------------------------
------------------------

im = standarizeImage(im) ;
width = size(im,2) ;
height = size(im,1) ;
numWords = size(model.vocab, 2) ;

% get PHOW features
[frames, descrs] = vl_phow(im,
model.phowOpts{:}) ;

% quantize local descriptors into visual
words
switch model.quantizer
  case 'vq'
    [drop, binsa] =
min(vl_alldist(model.vocab,
single(descrs)), [], 1) ;
  case 'kdtree'
    binsa =
double(vl_kdtreequery(model.kdtree,
model.vocab, ...
                          single(descrs), ...
                          'MaxComparisons',
50)) ;
end

for i = 1:length(model.numSpatialX)
  binsx =
vl_binsearch(linspace(1,width,model.num
SpatialX(i)+1), frames(1,:)) ;
  binsy =
vl_binsearch(linspace(1,height,model.num
SpatialY(i)+1), frames(2,:)) ;

  % combined quantization
  bins = sub2ind([model.numSpatialY(i),
model.numSpatialX(i), numWords], ...
```

```
                binsy,binsx,binsa) ;
  hist = zeros(model.numSpatialY(i) *
model.numSpatialX(i) * numWords, 1) ;
  hist = vl_binsum(hist, ones(size(bins)),
bins) ;
  hists{i} = single(hist / sum(hist)) ;
end
hist = cat(1,hists{:}) ;
hist = hist / sum(hist) ;

% ----------------------------------------------
-------------------------
function [className, score] =
classify(model, im)
% ----------------------------------------------
-------------------------

hist = getImageDescriptor(model, im) ;
psix = vl_homkermap(hist, 1, 'kchi2',
'gamma', .5) ;
scores = model.w' * psix + model.b' ;
[score, best] = max(scores) ;
className = model.classes{best} ;
```

Get training data

```
clear;
clc;
close all;

directory =
'Images/Samples/Processed/Cu';
fileNames = dir(directory);
fileNames(1:2) = [];

results = [];
for m = 1:length(fileNames)
   img =
imread(strcat(directory,'/',fileNames(m).na
me));
   result = TextureFeatureExtraction(img);
   results = [results; result];
   display(m);
   close all;
end
```

Train Data

```
function svmStructure =  TrainData()

[allNumber,allText,all] =
xlsread('Data/ClassificationDataNew.xlsx'
);
datas = allNumber(:,1:9)';
groups = all(:,10);

[datas(1,:), csMean] =
mapminmax(datas(1,:), 0, 1);
[datas(2,:), csStandardDeviation] =
mapminmax(datas(2,:), 0, 1);
[datas(3,:), csContrast] =
mapminmax(datas(3,:), 0, 1);

datas = datas';
svmStructure = svmtrain(datas,groups);
save('Data/svmStructure.mat',
'svmStructure');

save('Data/csMean.mat', 'csMean');
save('Data/csStandardDeviation.mat',
'csStandardDeviation');
save('Data/csContrast.mat', 'csContrast');
end
```

Texture Feature Extraction

```
function result =
TextureFeatureExtraction(I)

display('Extracting Features... Please
wait...');

format long g;
format compact;
fontSize = 18;


figure;
[rows, columns, numberOfColorBands] =
size(I);
```

```
subplot(2, 4, 1);
imshow(I);
title('Original Image', 'FontSize', fontSize);
set(gcf,
'units','normalized','outerposition',[0 0 1
1]);

% Extract the individual red, green, and
blue color channels.
redChannel = I(:, :, 1);
greenChannel = I(:, :, 2);
blueChannel = I(:, :, 3);

% Display the image.

subplot(2, 4, 2);
imshow(redChannel);
axis on;
title('Red Channel', 'FontSize', fontSize);

subplot(2, 4, 3);
imshow(greenChannel);
axis on;
title('Green Channel', 'FontSize', fontSize);

subplot(2, 4, 4);
imshow(blueChannel);
axis on;
title('Blue Channel', 'FontSize', fontSize);

% Threshold so we can get a binary image
to use as a mask.
blueChannel = blueChannel < 100;
greenChannel = greenChannel < 100;
redChannel = redChannel < 100;

binaryImage =
(redChannel&greenChannel&blueChannel
);
% Display the image.
subplot(2, 4, 5);
imshow(binaryImage);
axis on;
title('Binary Image', 'FontSize', fontSize);

% Fill to get rid of holes
binaryImage =
imcomplement(binaryImage);
binaryImage = imfill(binaryImage,
'holes');
% Get rid of small things
mask = bwareaopen(binaryImage, 1000);

maskedRgbImage = bsxfun(@times, I,
cast(mask, class(I)));
subplot(2, 4, 6);
imshow(maskedRgbImage);
axis on;

title('Masked Color Image', 'FontSize',
fontSize);

grayImage = rgb2gray(maskedRgbImage);
cc = bwconncomp(mask, 8);
mosquitoData =
regionprops(cc,grayImage,'all','MeanInten
sity');

imshow(grayImage);

title('Pre-processed', 'FontSize', fontSize);

mosquitoObject.MeanIntensity =
mean2(grayImage);
mosquitoObject.Variance =
var(double(grayImage));
mosquitoObject.StandardDeviation =
std2(double(grayImage));

[glcm0,s0] = graycomatrix(grayImage,
'offset', [0 1], 'Symmetric', false);
[glcm45,s45] = graycomatrix(grayImage,
'offset', [-1 1], 'Symmetric', false);
[glcm90,s90] =
graycomatrix(grayImage,'offset', [-1 0],
'Symmetric', false);
[glcm135,s135] =
graycomatrix(grayImage,'offset', [-1 -1],
'Symmetric', false);

base = 0.001;
```

```
glcmnorm0 = round((glcm0 ./
double(sum(glcm0(:)))./base)*base;
glcmnorm45 = round((glcm45 ./
double(sum(glcm45(:)))./base)*base;
glcmnorm90 = round((glcm90 ./
double(sum(glcm90(:)))./base)*base;
glcmnorm135 = round((glcm135 ./
double(sum(glcm135(:)))./base)*base;
mosquitoObject.glcm = glcmnorm0;
for m = 1:8
   for n = 1:8
      glvector = [glcmnorm0(m,n)
glcmnorm45(m,n) glcmnorm90(m,n)
glcmnorm135(m,n)];
      glcm(m,n) = mean(glvector);
   end
end

offset0 = [zeros(10,1) (1:10)'];
offset45 = [-(1:10)' (1:10)'];
offset90 = [-(1:10)' zeros(10,1)];
offset135 = [-(1:10)' -(1:10)'];

mosquitoObject.glcm =
graycomatrix(grayImage,'Offset',[offset0;o
ffset45;offset90;offset135]);
mosquitoObject.stats =
graycoprops(mosquitoObject.glcm,'Contra
st Correlation');
glcm_row = size(mosquitoObject.glcm,
1);
glcm_col = size(mosquitoObject.glcm, 2);
glcm_count = size(mosquitoObject.glcm,
3);

mosquitoObject.Autocorrelation =
zeros(1,glcm_count);
mosquitoObject.Contrast =
zeros(1,glcm_count);
mosquitoObject.CorrelationMatlab =
zeros(1,glcm_count);
mosquitoObject.CorrelationPaper =
zeros(1,glcm_count);
mosquitoObject.ClusterProminence =
zeros(1,glcm_count);
mosquitoObject.ClusterShade =
zeros(1,glcm_count);
mosquitoObject.Dissimilarity =
zeros(1,glcm_count);
mosquitoObject.Energy =
zeros(1,glcm_count);
mosquitoObject.Entropy =
zeros(1,glcm_count);
mosquitoObject.HomogeneityMatlab =
zeros(1,glcm_count);
mosquitoObject.HomogeneityPaper =
zeros(1,glcm_count);
mosquitoObject.MaximumProbability =
zeros(1,glcm_count);
mosquitoObject.SumOfSqaures =
zeros(1,glcm_count);
mosquitoObject.SumAverage =
zeros(1,glcm_count);
mosquitoObject.SumVariance =
zeros(1,glcm_count);
mosquitoObject.SumEntropy =
zeros(1,glcm_count);
mosquitoObject.DifferenceVariance =
zeros(1,glcm_count);
mosquitoObject.DifferenceEntropy =
zeros(1,glcm_count);
mosquitoObject.InformationMeasureCorre
lation1 = zeros(1,glcm_count);
mosquitoObject.InformationMeasureCorre
lation2 = zeros(1,glcm_count);
mosquitoObject.InverseDifferenceNormali
zed = zeros(1,glcm_count);
mosquitoObject.InverseDifferenceMoment
Normalized = zeros(1,glcm_count);

glcm_sum  = zeros(glcm_count,1);
glcm_mean = zeros(glcm_count,1);
glcm_var  = zeros(glcm_count,1);
u_x = zeros(glcm_count,1);
u_y = zeros(glcm_count,1);
s_x = zeros(glcm_count,1);
s_y = zeros(glcm_count,1);

p_x = zeros(glcm_row,glcm_count);
p_y = zeros(glcm_col,glcm_count);
```

```
p_xplusy = zeros((glcm_row*2 -
1),glcm_count);
p_xminusy =
zeros((glcm_row),glcm_count);
hxy  = zeros(glcm_count,1);
hxy1 = zeros(glcm_count,1);
hx   = zeros(glcm_count,1);
hy   = zeros(glcm_count,1);
hxy2 = zeros(glcm_count,1);

for k = 1: glcm_count
   glcm_sum(k) =
sum(sum(mosquitoObject.glcm(:,:,k)));
   mosquitoObject.glcm(:,:,k) =
mosquitoObject.glcm(:,:,k)./glcm_sum(k);
   glcm_mean(k) =
mean2(mosquitoObject.glcm(:,:,k));
   glcm_var(k)  =
(std2(mosquitoObject.glcm(:,:,k)))^2;
   for i = 1:glcm_row
      for j = 1:glcm_col
         mosquitoObject.Contrast(k) =
mosquitoObject.Contrast(k) + (abs(i -
j))^2.*mosquitoObject.glcm(i,j,k);
         mosquitoObject.Dissimilarity(k) =
mosquitoObject.Dissimilarity(k) + (abs(i -
j)*mosquitoObject.glcm(i,j,k));
         mosquitoObject.Energy(k) =
mosquitoObject.Energy(k) +
(mosquitoObject.glcm(i,j,k).^2);
         mosquitoObject.Entropy(k) =
mosquitoObject.Entropy(k) -
(mosquitoObject.glcm(i,j,k)*log(mosquito
Object.glcm(i,j,k) + eps));

mosquitoObject.HomogeneityMatlab(k) =
mosquitoObject.HomogeneityMatlab(k) +
(mosquitoObject.glcm(i,j,k)/( 1 + abs(i-j)
));

mosquitoObject.HomogeneityPaper(k) =
mosquitoObject.HomogeneityPaper(k) +
(mosquitoObject.glcm(i,j,k)/( 1 + (i -
j)^2));
         mosquitoObject.SumOfSqaures(k)
= mosquitoObject.SumOfSqaures(k) +
mosquitoObject.glcm(i,j,k)*((i -
glcm_mean(k))^2);


mosquitoObject.InverseDifferenceNormali
zed(k) =
mosquitoObject.InverseDifferenceNormali
zed(k) + (mosquitoObject.glcm(i,j,k)/( 1 +
(abs(i-j)/glcm_row) ));

mosquitoObject.InverseDifferenceMoment
Normalized(k) =
mosquitoObject.InverseDifferenceMoment
Normalized(k) +
(mosquitoObject.glcm(i,j,k)/( 1 + ((i -
j)/glcm_row)^2));
         u_x(k) = u_x(k) +
(i)*mosquitoObject.glcm(i,j,k);
         u_y(k) = u_y(k) +
(j)*mosquitoObject.glcm(i,j,k);
      end
   end


mosquitoObject.MaximumProbability(k) =
max(max(mosquitoObject.glcm(:,:,k)));
end

for k = 1:glcm_count
   for i = 1:glcm_row
      for j = 1:glcm_col
         p_x(i,k) = p_x(i,k) +
mosquitoObject.glcm(i,j,k);
         p_y(i,k) = p_y(i,k) +
mosquitoObject.glcm(j,i,k);
         if (ismember((i +
j),[2:2*glcm_row]))
            p_xplusy((i+j)-1,k) =
p_xplusy((i+j)-1,k) +
mosquitoObject.glcm(i,j,k);
         end
         if (ismember(abs(i-
j),[0:(glcm_row-1)]))
            p_xminusy((abs(i-j))+1,k) =
p_xminusy((abs(i-j))+1,k) +...
               mosquitoObject.glcm(i,j,k);
```

```
            end
        end
    end
end
for k = 1:(glcm_count)
    for i = 1:(2*(glcm_row)-1)
        mosquitoObject.SumAverage(k) =
mosquitoObject.SumAverage(k) +
(i+1)*p_xplusy(i,k);
        mosquitoObject.SumEntropy(k) =
mosquitoObject.SumEntropy(k) -
(p_xplusy(i,k)*log(p_xplusy(i,k) + eps));
    end
end

for k = 1:(glcm_count)
    for i = 1:(2*(glcm_row)-1)
        mosquitoObject.SumVariance(k) =
mosquitoObject.SumVariance(k) + (((i+1)
-
mosquitoObject.SumEntropy(k))^2)*p_xp
lusy(i,k);
    end
end

for k = 1:glcm_count
    for i = 0:(glcm_row-1)

mosquitoObject.DifferenceEntropy(k) =
mosquitoObject.DifferenceEntropy(k) -
(p_xminusy(i+1,k)*log(p_xminusy(i+1,k)
+ eps));

mosquitoObject.DifferenceVariance(k) =
mosquitoObject.DifferenceVariance(k) +
(i^2)*p_xminusy(i+1,k);
    end
end

for k = 1:glcm_count
    hxy(k) = mosquitoObject.Entropy(k);
    for i = 1:glcm_row
        for j = 1:glcm_col
            hxy1(k) = hxy1(k) -
(mosquitoObject.glcm(i,j,k)*log(p_x(i,k)*
p_y(j,k) + eps));
            hxy2(k) = hxy2(k) -
(p_x(i,k)*p_y(j,k)*log(p_x(i,k)*p_y(j,k) +
eps));
        end
        hx(k) = hx(k) - (p_x(i,k)*log(p_x(i,k)
+ eps));
        hy(k) = hy(k) - (p_y(i,k)*log(p_y(i,k)
+ eps));
    end

mosquitoObject.InformationMeasureCorre
lation1(k) = ( hxy(k) - hxy1(k) ) / (
max([hx(k),hy(k)]) );

mosquitoObject.InformationMeasureCorre
lation2(k) = ( 1 - exp( -2*( hxy2(k) -
hxy(k) ) ) )^0.5;
end

corm = zeros(glcm_count,1);
corp = zeros(glcm_count,1);
for k = 1:glcm_count
    for i = 1:glcm_row
        for j = 1:glcm_col
            s_x(k)  = s_x(k)  + (((i) -
u_x(k))^2)*mosquitoObject.glcm(i,j,k);
            s_y(k)  = s_y(k)  + (((j) -
u_y(k))^2)*mosquitoObject.glcm(i,j,k);
            corp(k) = corp(k) +
((i)*(j)*mosquitoObject.glcm(i,j,k));
            corm(k) = corm(k) + (((i) -
u_x(k))*((j) -
u_y(k))*mosquitoObject.glcm(i,j,k));

mosquitoObject.ClusterProminence(k) =
mosquitoObject.ClusterProminence(k) +
(((i + j - u_x(k) - u_y(k))^4)*...
                mosquitoObject.glcm(i,j,k));
            mosquitoObject.ClusterShade(k) =
mosquitoObject.ClusterShade(k) + (((i + j
- u_x(k) - u_y(k))^3)*...
                mosquitoObject.glcm(i,j,k));
        end
    end
    s_x(k) = s_x(k) ^ 0.5;
    s_y(k) = s_y(k) ^ 0.5;
```

```
    mosquitoObject.Autocorrelation(k) =
corp(k);
    mosquitoObject.CorrelationPaper(k) =
(corp(k) - u_x(k)*u_y(k))/(s_x(k)*s_y(k));
    mosquitoObject.CorrelationMatlab(k) =
corm(k) / (s_x(k)*s_y(k));
end

mosquitoObject.Autocorrelation =
[mean(mosquitoObject.Autocorrelation(1,
1:10)),
mean(mosquitoObject.Autocorrelation(1,1
1:20)),
mean(mosquitoObject.Autocorrelation(1,2
1:30)),mean(mosquitoObject.Autocorrelati
on(1,31:40))];
mosquitoObject.Contrast =
[mean(mosquitoObject.Contrast(1,1:10)),
mean(mosquitoObject.Contrast(1,11:20)),
mean(mosquitoObject.Contrast(1,21:30)),
mean(mosquitoObject.Contrast(1,31:40))];
mosquitoObject.CorrelationMatlab =
[mean(mosquitoObject.CorrelationMatlab(
1,1:10)),
mean(mosquitoObject.CorrelationMatlab(
1,11:20)),
mean(mosquitoObject.CorrelationMatlab(
1,21:30)),mean(mosquitoObject.Correlatio
nMatlab(1,31:40))];
mosquitoObject.CorrelationPaper =
[mean(mosquitoObject.CorrelationPaper(1
,1:10)),
mean(mosquitoObject.CorrelationPaper(1,
11:20)),
mean(mosquitoObject.CorrelationPaper(1,
21:30)),mean(mosquitoObject.Correlation
Paper(1,31:40))];
mosquitoObject.ClusterProminence =
[mean(mosquitoObject.ClusterProminence
(1,1:10)),
mean(mosquitoObject.ClusterProminence(
1,11:20)),
mean(mosquitoObject.ClusterProminence(
1,21:30)),mean(mosquitoObject.ClusterPr
ominence(1,31:40))];
mosquitoObject.ClusterShade =
[mean(mosquitoObject.ClusterShade(1,1:1
0)),
mean(mosquitoObject.ClusterShade(1,11:
20)),
mean(mosquitoObject.ClusterShade(1,21:
30)),mean(mosquitoObject.ClusterShade(1
,31:40))];
mosquitoObject.Dissimilarity =
[mean(mosquitoObject.Dissimilarity(1,1:1
0)),
mean(mosquitoObject.Dissimilarity(1,11:2
0)),
mean(mosquitoObject.Dissimilarity(1,21:3
0)),mean(mosquitoObject.Dissimilarity(1,
31:40))];
mosquitoObject.Energy =
[mean(mosquitoObject.Energy(1,1:10)),
mean(mosquitoObject.Energy(1,11:20)),
mean(mosquitoObject.Energy(1,21:30)),m
ean(mosquitoObject.Energy(1,31:40))];
mosquitoObject.Entropy =
[mean(mosquitoObject.Entropy(1,1:10)),
mean(mosquitoObject.Entropy(1,11:20)),
mean(mosquitoObject.Entropy(1,21:30)),
mean(mosquitoObject.Entropy(1,31:40))];
mosquitoObject.HomogeneityMatlab =
[mean(mosquitoObject.HomogeneityMatla
b(1,1:10)),
mean(mosquitoObject.HomogeneityMatla
b(1,11:20)),
mean(mosquitoObject.HomogeneityMatla
b(1,21:30)),mean(mosquitoObject.Homog
eneityMatlab(1,31:40))];
mosquitoObject.HomogeneityPaper =
[mean(mosquitoObject.HomogeneityPaper
(1,1:10)),
mean(mosquitoObject.HomogeneityPaper(
1,11:20)),
mean(mosquitoObject.HomogeneityPaper(
1,21:30)),mean(mosquitoObject.Homogen
eityPaper(1,31:40))];
mosquitoObject.MaximumProbability =
[mean(mosquitoObject.MaximumProbabil
ity(1,1:10)),
mean(mosquitoObject.MaximumProbabilit
```

```
y(1,11:20)),
mean(mosquitoObject.MaximumProbabilit
y(1,21:30)),mean(mosquitoObject.Maxim
umProbability(1,31:40))];

mosquitoObject.SumOfSqaures =
[mean(mosquitoObject.SumOfSqaures(1,1
:10)),
mean(mosquitoObject.SumOfSqaures(1,1
1:20)),
mean(mosquitoObject.SumOfSqaures(1,2
1:30)),mean(mosquitoObject.SumOfSqaur
es(1,31:40))];
mosquitoObject.SumAverage =
[mean(mosquitoObject.SumAverage(1,1:1
0)),
mean(mosquitoObject.SumAverage(1,11:2
0)),
mean(mosquitoObject.SumAverage(1,21:3
0)),mean(mosquitoObject.SumAverage(1,
31:40))];
mosquitoObject.SumVariance =
[mean(mosquitoObject.SumVariance(1,1:
10)),
mean(mosquitoObject.SumVariance(1,11:
20)),
mean(mosquitoObject.SumVariance(1,21:
30)),mean(mosquitoObject.SumVariance(
1,31:40))];
mosquitoObject.SumEntropy =
[mean(mosquitoObject.SumEntropy(1,1:1
0)),
mean(mosquitoObject.SumEntropy(1,11:2
0)),
mean(mosquitoObject.SumEntropy(1,21:3
0)),mean(mosquitoObject.SumEntropy(1,3
1:40))];
mosquitoObject.DifferenceVariance =
[mean(mosquitoObject.DifferenceVarianc
e(1,1:10)),
mean(mosquitoObject.DifferenceVariance
(1,11:20)),
mean(mosquitoObject.DifferenceVariance
(1,21:30)),mean(mosquitoObject.Differenc
eVariance(1,31:40))];
mosquitoObject.DifferenceEntropy =
[mean(mosquitoObject.DifferenceEntropy
(1,1:10)),
mean(mosquitoObject.DifferenceEntropy(
1,11:20)),
mean(mosquitoObject.DifferenceEntropy(
1,21:30)),mean(mosquitoObject.Differenc
eEntropy(1,31:40))];
mosquitoObject.InformationMeasureCorre
lation1 =
[mean(mosquitoObject.InformationMeasur
eCorrelation1(1,1:10)),
mean(mosquitoObject.InformationMeasur
eCorrelation1(1,11:20)),
mean(mosquitoObject.InformationMeasur
eCorrelation1(1,21:30)),mean(mosquitoOb
ject.InformationMeasureCorrelation1(1,31
:40))];
mosquitoObject.InformationMeasureCorre
lation2 =
[mean(mosquitoObject.InformationMeasur
eCorrelation2(1,1:10)),
mean(mosquitoObject.InformationMeasur
eCorrelation2(1,11:20)),
mean(mosquitoObject.InformationMeasur
eCorrelation2(1,21:30)),mean(mosquitoOb
ject.InformationMeasureCorrelation2(1,31
:40))];
mosquitoObject.InverseDifferenceNormali
zed =
[mean(mosquitoObject.InverseDifference
Normalized(1,1:10)),
mean(mosquitoObject.InverseDifferenceN
ormalized(1,11:20)),
mean(mosquitoObject.InverseDifferenceN
ormalized(1,21:30)),mean(mosquitoObject
.InverseDifferenceNormalized(1,31:40))];
mosquitoObject.InverseDifferenceMoment
Normalized =
[mean(mosquitoObject.InverseDifference
MomentNormalized(1,1:10)),
mean(mosquitoObject.InverseDifferenceM
omentNormalized(1,11:20)),
mean(mosquitoObject.InverseDifferenceM
omentNormalized(1,21:30)),mean(mosquit
```

```
oObject.InverseDifferenceMomentNormal
ized(1,31:40))];


result = [];
result = [mosquitoObject.MeanIntensity;
mosquitoObject.MeanIntensity;
mosquitoObject.MeanIntensity;
mosquitoObject.MeanIntensity];
result = horzcat(result,
[mosquitoObject.StandardDeviation;
mosquitoObject.StandardDeviation;
mosquitoObject.StandardDeviation;
mosquitoObject.StandardDeviation]);
result = horzcat(result,
mosquitoObject.Contrast');
result = horzcat(result,
mosquitoObject.CorrelationMatlab');
result = horzcat(result,
mosquitoObject.Dissimilarity');
result = horzcat(result,
mosquitoObject.Energy');
result = horzcat(result,
mosquitoObject.Entropy');
result = horzcat(result,
mosquitoObject.HomogeneityMatlab');
result = horzcat(result,
mosquitoObject.MaximumProbability');

assignin('base','result',result);

display('Done Extracting Features...');

end
```

Preprocessing

```
function preProcessedImage =
Preprocessing(rawImage)

display('Processing the original image...
Please wait...');
tic;
I = rawImage;
IGray = rgb2gray(I);
mask = zeros(size(IGray));
mask(2:end-2,2:end-2) = 1;

numIter = 5000;
bw = activecontour(IGray, mask,
numIter);
Iuint8 = im2uint8(bw);
result = imsubtract(Iuint8,IGray);

[row, column] = size(result);

for m = 1:row
   for n = 1:column
      if(result(m,n) <= 140)
         I(m,n,:) = 255 .* 255;
      end
   end
end

preProcessedImage = I;
display('Done Preprocessing the Image...');
toc;
end
```

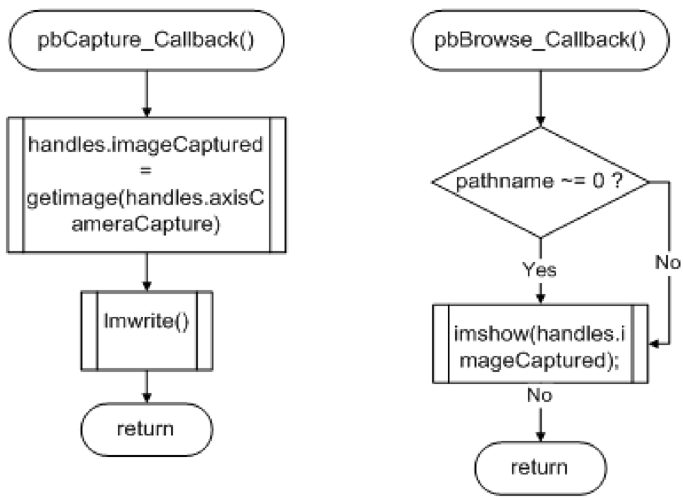
Testing Accuracy

```
clear;
clc;
close all;

directory =
'Images/Samples/Processed/AAl';
fileNames = dir(directory);
fileNames(1:2) = [];

results = [];
for m = 1:length(fileNames)

  close all;

  img =
imread(strcat(directory,'/',fileNames(m).na
me));
```

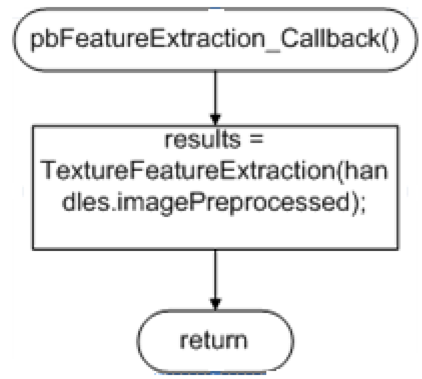
```
  result = TextureFeatureExtraction(img);
  load('Data/svmStructure.mat');
  load('Data/csMean.mat');
  load('Data/csStandardDeviation.mat');
  load('Data/csContrast.mat');

  result(:,1) = mapminmax('apply',
result(:,1), csMean);
  result(:,2) = mapminmax('apply',
result(:,2), csStandardDeviation);
  result(:,3) = mapminmax('apply',
result(:,3), csContrast);

  species =
svmclassify(svmStructure,result);

  display(m);
  display(species);
  results = [results; species];
end
```

APPENDIX B

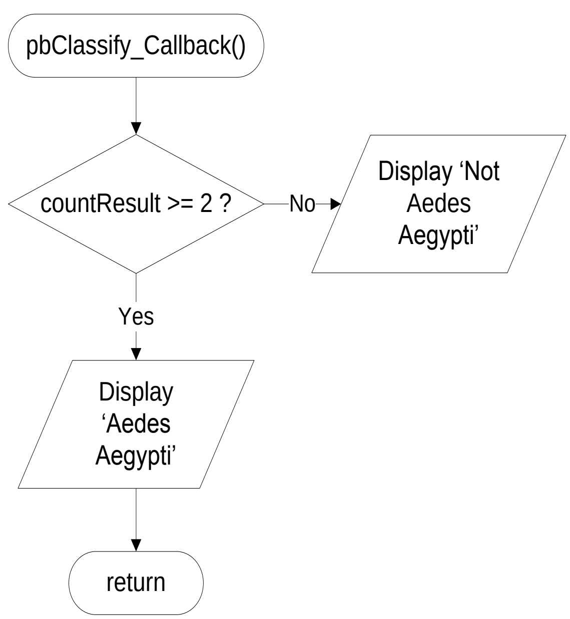
Get Image Flowchart



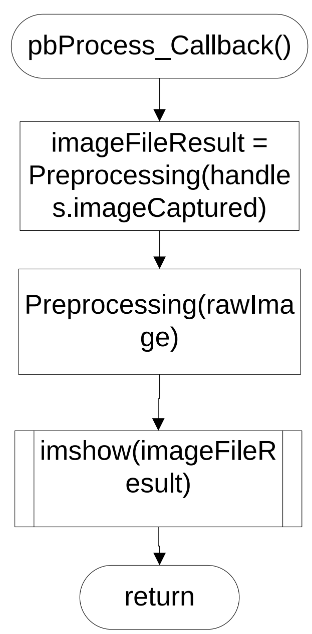
Extract Image Flowchart



Classify Image Flowchart



Process Image Flowchart



APPENDIX C

Mean	Standard Dev.	Contrast	Correlation	Dissimilarity	Energy	Entropy	Homogeneity	Max. Probability	Mosquito
239.0226	54.97270902	1.521406	0.699226071	0.287919986	0.81224	0.618652	0.945933305	0.900877583	Aedes Aegypti
239.0226	54.97270902	1.81986	0.644084824	0.337826408	0.802819	0.645111	0.938866262	0.895593099	Aedes Aegypti
239.0226	54.97270902	1.523553	0.699421871	0.290181264	0.811483	0.620791	0.945315942	0.900453322	Aedes Aegypti
239.0226	54.97270902	1.5724	0.692647918	0.294812198	0.809287	0.625608	0.945301004	0.899214376	Aedes Aegypti
236.9894	59.95752208	1.842773	0.693886342	0.322903256	0.800084	0.627797	0.944428558	0.893821003	Aedes Aegypti
236.9894	59.95752208	1.905054	0.687370428	0.333281222	0.795767	0.638008	0.943464198	0.891368318	Aedes Aegypti
236.9894	59.95752208	1.699565	0.718523572	0.302192496	0.80203	0.62128	0.947199396	0.89490552	Aedes Aegypti
236.9894	59.95752208	2.120243	0.652006426	0.367595797	0.792272	0.650126	0.937730427	0.889415816	Aedes Aegypti
237.1556	59.00274677	1.982913	0.654818836	0.355143743	0.790899	0.666308	0.938581396	0.88868098	Aedes Aegypti
237.1556	59.00274677	2.246218	0.613599247	0.399682015	0.782903	0.689623	0.931825919	0.884147531	Aedes Aegypti
237.1556	59.00274677	1.943295	0.662579008	0.353187518	0.790338	0.668828	0.938046854	0.888387414	Aedes Aegypti
237.1556	59.00274677	2.350146	0.595757019	0.413348373	0.780656	0.693308	0.930575974	0.882829997	Aedes Aegypti
237.5214	59.00227621	1.805489	0.690288505	0.318538763	0.804285	0.619524	0.944651652	0.896228179	Aedes Aegypti
237.5214	59.00227621	1.94007	0.671341291	0.339716551	0.79862	0.633961	0.942214292	0.893027324	Aedes Aegypti
237.5214	59.00227621	1.622912	0.722617028	0.290150693	0.806817	0.60973	0.949125148	0.897622211	Aedes Aegypti
237.5214	59.00227621	2.063329	0.650456217	0.359669198	0.796705	0.640625	0.938813177	0.891952015	Aedes Aegypti
236.5657	60.09195486	2.102619	0.645979411	0.375382964	0.784651	0.682832	0.935472779	0.885108585	Aedes Aegypti
236.5657	60.09195486	2.366294	0.605740855	0.418296734	0.777243	0.704074	0.929126545	0.880894261	Aedes Aegypti
236.5657	60.09195486	2.038463	0.657214558	0.369107649	0.784846	0.683213	0.935626972	0.88525111	Aedes Aegypti
236.5657	60.09195486	2.46769	0.588885893	0.43252669	0.774644	0.70866	0.927789756	0.879370326	Aedes Aegypti
242.0746	51.61097169	1.37149	0.691293995	0.239807992	0.853027	0.478654	0.959282352	0.923249145	Aedes Aegypti
242.0746	51.61097169	1.6527	0.632236813	0.283930743	0.845142	0.500581	0.953259773	0.918948246	Aedes Aegypti
242.0746	51.61097169	1.255331	0.717909665	0.223064907	0.855384	0.47253	0.961069041	0.924538423	Aedes Aegypti
242.0746	51.61097169	1.528446	0.659909191	0.264393806	0.847733	0.493138	0.955915985	0.920362821	Aedes Aegypti
235.9078	60.80449334	1.931018	0.683388054	0.351296098	0.78125	0.697565	0.937684767	0.883204402	Aedes Aegypti
235.9078	60.80449334	2.470184	0.599530754	0.437600508	0.767228	0.73598	0.925643134	0.875156215	Aedes Aegypti
235.9078	60.80449334	2.022943	0.669093536	0.36666567	0.778579	0.705734	0.935334692	0.881693464	Aedes Aegypti

235.9078	60.80449334	2.494515	0.595604501	0.441156133	0.766729	0.737777	0.925232766	0.874866626	Aedes Aegypti
240.9566	53.44602519	1.630683	0.658147504	0.285207169	0.836609	0.526592	0.951882135	0.914242925	Aedes Aegypti
240.9566	53.44602519	1.887657	0.609407689	0.325385833	0.829092	0.546916	0.946522258	0.910095652	Aedes Aegypti
240.9566	53.44602519	1.479995	0.690611358	0.261614325	0.839356	0.518532	0.955228833	0.915753062	Aedes Aegypti
240.9566	53.44602519	1.772205	0.633315575	0.306908876	0.831673	0.540306	0.94914327	0.91152436	Aedes Aegypti
235.6137	61.37206205	1.81008	0.707876419	0.331344005	0.782525	0.694974	0.940187338	0.8839344	Aedes Aegypti
235.6137	61.37206205	2.384123	0.619115246	0.422967974	0.767458	0.736492	0.927594986	0.875294214	Aedes Aegypti
235.6137	61.37206205	2.061732	0.667837827	0.368973237	0.776424	0.710618	0.935608547	0.880427971	Aedes Aegypti
235.6137	61.37206205	2.417486	0.613851593	0.426670241	0.767226	0.737576	0.927323392	0.875152454	Aedes Aegypti
239.0734	56.66429413	1.860378	0.652536541	0.324181429	0.816324	0.584174	0.945313993	0.902967241	Aedes Aegypti
239.0734	56.66429413	2.120237	0.610093342	0.365173243	0.807946	0.605781	0.939871126	0.898267353	Aedes Aegypti
239.0734	56.66429413	1.696463	0.684891942	0.298947809	0.818559	0.576814	0.949093119	0.904200051	Aedes Aegypti
239.0734	56.66429413	2.098868	0.613969458	0.361967353	0.808985	0.60452	0.940052367	0.898865461	Aedes Aegypti
232.6376	69.33289318	2.406183	0.676250275	0.391528037	0.77029	0.62924	0.941591191	0.875477967	Aedes Aegypti
232.6376	69.33289318	2.805682	0.626446231	0.450493509	0.761103	0.654099	0.933858702	0.870220412	Aedes Aegypti
232.6376	69.33289318	2.333366	0.686277746	0.382179794	0.770688	0.627075	0.942794542	0.875713554	Aedes Aegypti
232.6376	69.33289318	2.922593	0.610894714	0.467055349	0.758343	0.658376	0.932126674	0.868574906	Aedes Aegypti
238.9378	55.77037227	1.627498	0.687169913	0.297283695	0.814394	0.605429	0.946058751	0.902034909	Aedes Aegypti
238.9378	55.77037227	1.72346	0.673739808	0.311977417	0.810228	0.617356	0.944125328	0.899702857	Aedes Aegypti
238.9378	55.77037227	1.524635	0.708528835	0.27903604	0.815679	0.599495	0.949125734	0.902737274	Aedes Aegypti
238.9378	55.77037227	1.849814	0.649692304	0.332646734	0.806979	0.626263	0.94127313	0.897883081	Aedes Aegypti
235.9957	60.81999157	2.17864	0.642995668	0.390017076	0.777532	0.702711	0.932971353	0.881038072	Aedes Aegypti
235.9957	60.81999157	2.54659	0.58760705	0.445617791	0.768622	0.727045	0.925421045	0.875902155	Aedes Aegypti
235.9957	60.81999157	2.274957	0.627988701	0.406200876	0.774601	0.710578	0.930516068	0.879368191	Aedes Aegypti
235.9957	60.81999157	2.465647	0.600661385	0.435077182	0.769448	0.724815	0.926649912	0.876393484	Aedes Aegypti
241.4767	51.5765263	1.379751	0.690052941	0.255532746	0.84201	0.528068	0.953069799	0.917342114	Aedes Aegypti
241.4767	51.5765263	1.393619	0.6912779	0.25715332	0.839373	0.535076	0.953186656	0.915896344	Aedes Aegypti
241.4767	51.5765263	1.396868	0.687764202	0.257326861	0.841111	0.530533	0.953013251	0.916848761	Aedes Aegypti
241.4767	51.5765263	1.64784	0.635026447	0.2980579	0.833684	0.55117	0.947419563	0.912757217	Aedes Aegypti
235.2873	61.77766525	2.102109	0.66583103	0.377332131	0.77317	0.719062	0.934138176	0.878544987	Aedes Aegypti
235.2873	61.77766525	2.600611	0.590994588	0.458302522	0.75997	0.75501	0.922571186	0.870929899	Aedes Aegypti

235.2873	61.77766525	2.128645	0.662096701	0.385032137	0.771265	0.7248	0.932463089	0.877472053	Aedes Aegypti
235.2873	61.77766525	2.687903	0.577329961	0.470632434	0.75834	0.758749	0.921145178	0.869958913	Aedes Aegypti
240.7888	53.08147624	1.357905	0.711681517	0.250512796	0.838059	0.537236	0.953945466	0.915155289	Aedes Aegypti
240.7888	53.08147624	1.524573	0.680679949	0.275092648	0.832438	0.553087	0.9509546	0.912058695	Aedes Aegypti
240.7888	53.08147624	1.399378	0.704188512	0.257762574	0.836513	0.542096	0.952707472	0.914308639	Aedes Aegypti
240.7888	53.08147624	1.558236	0.673642851	0.284636558	0.831011	0.557955	0.948840363	0.911278888	Aedes Aegypti
235.6095	61.36600787	1.850515	0.701584251	0.336929502	0.781648	0.696069	0.939574596	0.883429704	Aedes Aegypti
235.6095	61.36600787	2.402085	0.616919005	0.425408088	0.766898	0.736605	0.927315616	0.87496506	Aedes Aegypti
235.6095	61.36600787	2.103583	0.661832985	0.375232024	0.775339	0.712041	0.934833748	0.879794333	Aedes Aegypti
235.6095	61.36600787	2.423701	0.613545947	0.426360157	0.766985	0.736548	0.927637474	0.875003269	Aedes Aegypti
238.3364	57.51630711	1.501544	0.728989447	0.273987084	0.816424	0.591592	0.950018644	0.903110599	Aedes Aegypti
238.3364	57.51630711	1.784929	0.682386097	0.316774321	0.808042	0.615342	0.944634776	0.898426421	Aedes Aegypti
238.3364	57.51630711	1.626736	0.707822859	0.294428718	0.812764	0.601776	0.946971536	0.901066328	Aedes Aegypti
238.3364	57.51630711	1.773955	0.684466785	0.320139748	0.80742	0.617157	0.943005652	0.898079252	Aedes Aegypti
234.7615	62.39754855	2.196943	0.657034217	0.394602115	0.766527	0.741081	0.93066991	0.874747429	Aedes Aegypti
234.7615	62.39754855	2.794564	0.568675811	0.486202692	0.752287	0.778933	0.918447871	0.866449985	Aedes Aegypti
234.7615	62.39754855	2.290032	0.643417949	0.407759545	0.764032	0.747589	0.929303229	0.873298432	Aedes Aegypti
234.7615	62.39754855	2.402834	0.629264465	0.429508665	0.75953	0.761473	0.925586801	0.870706779	Aedes Aegypti
241.1827	52.61590751	1.299501	0.719919314	0.235267653	0.843671	0.513548	0.95800328	0.918193912	Aedes Aegypti
241.1827	52.61590751	1.444196	0.692878603	0.261580849	0.838509	0.530185	0.953477741	0.91537685	Aedes Aegypti
241.1827	52.61590751	1.319623	0.716811822	0.240074568	0.843473	0.515849	0.956557567	0.918094813	Aedes Aegypti
241.1827	52.61590751	1.590553	0.661696201	0.282492711	0.83574	0.53812	0.950943941	0.913852206	Aedes Aegypti
234.9041	62.06392876	2.228992	0.648902702	0.401277418	0.766628	0.744082	0.929068675	0.874837086	Aedes Aegypti
234.9041	62.06392876	2.739197	0.573333367	0.481461119	0.753116	0.779891	0.918487729	0.866978213	Aedes Aegypti
234.9041	62.06392876	2.15488	0.661636451	0.393944263	0.766548	0.745091	0.929656826	0.874800661	Aedes Aegypti
234.9041	62.06392876	2.610729	0.593486891	0.465029318	0.755489	0.773952	0.919996689	0.868360744	Aedes Aegypti
235.0279	61.74530278	2.099115	0.665772951	0.378624471	0.769307	0.735961	0.932726629	0.876382892	Aedes Aegypti
235.0279	61.74530278	2.314857	0.635948371	0.419022582	0.761202	0.759544	0.926145342	0.871701798	Aedes Aegypti
235.0279	61.74530278	1.870506	0.703271033	0.342117949	0.773575	0.724071	0.93800688	0.878853928	Aedes Aegypti
235.0279	61.74530278	2.592719	0.592070479	0.45352982	0.757324	0.768992	0.923016718	0.869430075	Aedes Aegypti
237.3815	59.66969198	1.742528	0.706397443	0.305660854	0.806385	0.606124	0.947284675	0.897344582	Aedes Aegypti

237.3815	59.66969198	2.13515	0.644651213	0.365394407	0.795898	0.633629	0.939661131	0.891414956	Aedes Aegypti
237.3815	59.66969198	1.541518	0.741059533	0.273328603	0.809793	0.594089	0.952439272	0.899225801	Aedes Aegypti
237.3815	59.66969198	1.976385	0.671045721	0.341078576	0.799491	0.625158	0.942764724	0.893461846	Aedes Aegypti
236.4149	59.90316686	1.847331	0.687842826	0.334982452	0.786634	0.684255	0.940481066	0.886299487	Aedes Aegypti
236.4149	59.90316686	2.364023	0.605005146	0.419731108	0.772604	0.722621	0.92829484	0.878262481	Aedes Aegypti
236.4149	59.90316686	1.90823	0.678306838	0.346964232	0.78423	0.690595	0.938288634	0.884930636	Aedes Aegypti
236.4149	59.90316686	2.362523	0.605279757	0.418540609	0.77287	0.722621	0.928595376	0.878417375	Aedes Aegypti
233.6009	68.05722631	2.216494	0.688704706	0.360736942	0.782096	0.595614	0.946014463	0.882388471	Aedes Aegypti
233.6009	68.05722631	2.674709	0.627947811	0.428137725	0.77152	0.622067	0.937396711	0.876333688	Aedes Aegypti
233.6009	68.05722631	2.331958	0.672757494	0.377220134	0.779781	0.60083	0.94398084	0.881040412	Aedes Aegypti
233.6009	68.05722631	2.855986	0.602769452	0.454754044	0.768788	0.6293	0.933824553	0.87474972	Aedes Aegypti
235.6692	61.42619026	2.014774	0.675418779	0.363115169	0.779752	0.699166	0.936234597	0.88231897	Aedes Aegypti
235.6692	61.42619026	2.464179	0.607662665	0.43431088	0.767776	0.73199	0.926510882	0.875443061	Aedes Aegypti
235.6692	61.42619026	2.211852	0.644857462	0.394632435	0.774918	0.712944	0.931779101	0.879555263	Aedes Aegypti
235.6692	61.42619026	2.544275	0.594771676	0.446880163	0.766006	0.736692	0.924788502	0.874429667	Aedes Aegypti
236.3665	59.88268442	2.061182	0.652301321	0.375111751	0.779754	0.701586	0.934386837	0.88236916	Aedes Aegypti
236.3665	59.88268442	2.33786	0.610248737	0.420998882	0.77151	0.724632	0.927590721	0.877644206	Aedes Aegypti
236.3665	59.88268442	2.022839	0.659563365	0.373562124	0.778764	0.703669	0.933939106	0.881811981	Aedes Aegypti
236.3665	59.88268442	2.313945	0.614235865	0.416642745	0.771629	0.723188	0.928473113	0.877704462	Aedes Aegypti
233.7897	63.78154589	2.331652	0.652965815	0.421142666	0.75463	0.769624	0.926633504	0.867791413	Aedes Aegypti
233.7897	63.78154589	2.764224	0.593012751	0.49083646	0.743265	0.80139	0.916557705	0.861171756	Aedes Aegypti
233.7897	63.78154589	2.307975	0.65672886	0.421918179	0.753796	0.772913	0.925742752	0.867342175	Aedes Aegypti
233.7897	63.78154589	2.902632	0.572703141	0.512382233	0.739951	0.807981	0.913952304	0.859179742	Aedes Aegypti
240.6404	53.17868862	1.865874	0.599743802	0.330299127	0.823261	0.579234	0.943905186	0.906933186	Aedes Aegypti
240.6404	53.17868862	1.959074	0.586060097	0.344226669	0.819825	0.588585	0.941936783	0.905013548	Aedes Aegypti
240.6404	53.17868862	1.679238	0.641579615	0.302724285	0.825824	0.571521	0.947660877	0.908357067	Aedes Aegypti
240.6404	53.17868862	1.996465	0.577954239	0.351747648	0.817692	0.594259	0.941080733	0.90383897	Aedes Aegypti
237.4222	58.30909992	2.191488	0.610550444	0.390482251	0.787349	0.682634	0.932881574	0.886714885	Aedes Aegypti
237.4222	58.30909992	2.445935	0.571737453	0.428277051	0.780269	0.700994	0.927983742	0.882652981	Aedes Aegypti
237.4222	58.30909992	2.035614	0.63928042	0.366046328	0.789482	0.675882	0.93665265	0.887931167	Aedes Aegypti
237.4222	58.30909992	2.490462	0.563921345	0.438458381	0.77792	0.707319	0.926372883	0.881318145	Aedes Aegypti

237.8434	57.38213512	1.712782	0.687250257	0.305846869	0.802387	0.642535	0.945531504	0.895278764	Aedes Aegypti
237.8434	57.38213512	2.111477	0.619598778	0.373787438	0.790467	0.679998	0.935277453	0.888566796	Aedes Aegypti
237.8434	57.38213512	1.838204	0.665493147	0.331119375	0.797528	0.658848	0.94132884	0.892552713	Aedes Aegypti
237.8434	57.38213512	1.973834	0.644634574	0.351755573	0.792991	0.670766	0.938623929	0.889976021	Aedes Aegypti
238.9355	56.05200388	1.969205	0.623119736	0.347702132	0.806932	0.624796	0.940748983	0.897805417	Aedes Aegypti
238.9355	56.05200388	2.333856	0.559187405	0.406171105	0.796824	0.652088	0.932590487	0.89209774	Aedes Aegypti
238.9355	56.05200388	1.755955	0.664780186	0.31724484	0.810804	0.614892	0.944276189	0.899985301	Aedes Aegypti
238.9355	56.05200388	2.171638	0.589846565	0.383639093	0.799893	0.644465	0.934988364	0.893849582	Aedes Aegypti
242.0193	50.65558643	1.616357	0.618980529	0.288314717	0.840348	0.530918	0.950526638	0.916397101	Aedes Aegypti
242.0193	50.65558643	1.782206	0.585683331	0.315873106	0.834578	0.545921	0.946483235	0.913209193	Aedes Aegypti
242.0193	50.65558643	1.451418	0.659181598	0.264453828	0.843498	0.522243	0.953302953	0.918125775	Aedes Aegypti
242.0193	50.65558643	1.78766	0.58433382	0.3153916	0.834378	0.54611	0.946981001	0.913101546	Aedes Aegypti
239.7936	54.63192916	1.984782	0.596323943	0.350414359	0.813103	0.607635	0.94075556	0.901263795	Aedes Aegypti
239.7936	54.63192916	2.087083	0.582032304	0.366405953	0.809286	0.618178	0.938335875	0.899117924	Aedes Aegypti
239.7936	54.63192916	1.755561	0.644765589	0.317284184	0.816468	0.598174	0.944980526	0.903148332	Aedes Aegypti
239.7936	54.63192916	2.095905	0.580072195	0.369572603	0.807665	0.622303	0.938036114	0.898224679	Aedes Aegypti
233.6123	64.8613169	1.879703	0.733241771	0.331511622	0.771841	0.70032	0.942112722	0.877620966	Aedes Aegypti
233.6123	64.8613169	2.271949	0.681485521	0.395792349	0.761356	0.73279	0.932466696	0.871614434	Aedes Aegypti
233.6123	64.8613169	1.927461	0.727175886	0.343671191	0.770236	0.707588	0.939199973	0.876729752	Aedes Aegypti
233.6123	64.8613169	2.204869	0.690899608	0.386371538	0.7628	0.72949	0.933369224	0.872457613	Aedes Aegypti
231.7599	71.30492407	1.800978	0.769278813	0.292188033	0.782087	0.569831	0.956135277	0.881735163	Aedes Aegypti
231.7599	71.30492407	2.335857	0.703271554	0.371368009	0.769771	0.602752	0.946021245	0.874780393	Aedes Aegypti
231.7599	71.30492407	2.120461	0.728350645	0.338865052	0.775671	0.587181	0.950322551	0.878123845	Aedes Aegypti
231.7599	71.30492407	2.449852	0.688928973	0.386911238	0.76807	0.606935	0.944131825	0.873776123	Aedes Aegypti
237.5012	57.8907807	2.22319	0.598303382	0.399196696	0.784871	0.69584	0.930813101	0.885358365	Aedes Aegypti
237.5012	57.8907807	2.234785	0.601752262	0.404201071	0.782445	0.703935	0.92941182	0.883983765	Aedes Aegypti
237.5012	57.8907807	1.87995	0.661327442	0.347458403	0.791374	0.678275	0.937537317	0.889064454	Aedes Aegypti
237.5012	57.8907807	2.554223	0.544548871	0.452461653	0.775453	0.721163	0.923305078	0.879965201	Aedes Aegypti
240.4821	52.93617037	1.446697	0.687982918	0.270565671	0.827797	0.57637	0.950028477	0.909520249	Aedes Aegypti
240.4821	52.93617037	1.846054	0.607052992	0.330909082	0.816803	0.605353	0.94251661	0.903393263	Aedes Aegypti
240.4821	52.93617037	1.598737	0.656196122	0.295879733	0.823368	0.589516	0.946584472	0.90706745	Aedes Aegypti

240.4821	52.93617037	1.972302	0.580197533	0.354639532	0.813322	0.615961	0.938725055	0.901448641	Aedes Aegypti
237.0123	58.47072809	1.641968	0.710046002	0.306635343	0.793417	0.676758	0.94255556	0.890280022	Aedes Aegypti
237.0123	58.47072809	1.846829	0.678014015	0.343540778	0.785475	0.701089	0.936787348	0.885782991	Aedes Aegypti
237.0123	58.47072809	1.432913	0.747507362	0.276167834	0.796493	0.665785	0.946612431	0.892001971	Aedes Aegypti
237.0123	58.47072809	1.687302	0.705903746	0.31346564	0.789822	0.685565	0.94191873	0.88823305	Aedes Aegypti
240.7801	53.15419352	1.48051	0.682314921	0.265261514	0.834465	0.54774	0.953448839	0.913141725	Aedes Aegypti
240.7801	53.15419352	1.986978	0.579654288	0.346219242	0.820534	0.585868	0.942254661	0.905409334	Aedes Aegypti
240.7801	53.15419352	1.499309	0.679707563	0.274111313	0.831915	0.556085	0.951271965	0.911743382	Aedes Aegypti
240.7801	53.15419352	2.05785	0.564711085	0.357976015	0.819274	0.589256	0.94039232	0.904699997	Aedes Aegypti
238.9868	56.24069142	1.95779	0.625691734	0.346140249	0.809249	0.617696	0.940743725	0.899086024	Aedes Aegypti
238.9868	56.24069142	2.043111	0.615321858	0.361414852	0.804955	0.629828	0.938252881	0.896677488	Aedes Aegypti
238.9868	56.24069142	1.756119	0.665730415	0.317546108	0.811831	0.609596	0.944383551	0.900530784	Aedes Aegypti
238.9868	56.24069142	2.175164	0.590349826	0.381874427	0.801483	0.638071	0.935883408	0.894716827	Aedes Aegypti
238.2027	57.18372636	1.913216	0.645926491	0.344201047	0.800711	0.645851	0.939900865	0.894320635	Aedes Aegypti
238.2027	57.18372636	2.196669	0.599606552	0.386635882	0.792584	0.666931	0.934440275	0.889705104	Aedes Aegypti
238.2027	57.18372636	1.823395	0.663944423	0.331288051	0.80108	0.642899	0.941823044	0.894513423	Aedes Aegypti
238.2027	57.18372636	2.275073	0.58528909	0.402132007	0.789931	0.674216	0.931988162	0.888202165	Aedes Aegypti
235.6467	61.16391603	1.664774	0.73482434	0.306477187	0.787414	0.675959	0.943352047	0.886766596	Aedes Aegypti
235.6467	61.16391603	2.159747	0.660074027	0.385890145	0.774969	0.712946	0.931967879	0.879667673	Aedes Aegypti
235.6467	61.16391603	1.800827	0.713931537	0.329618796	0.784351	0.686836	0.939513391	0.88504149	Aedes Aegypti
235.6467	61.16391603	2.068114	0.674425959	0.371027654	0.776969	0.70763	0.933967904	0.880826717	Aedes Aegypti
237.5928	59.15850422	1.665702	0.714943145	0.292297479	0.809506	0.600297	0.949381243	0.899127538	Aedes Aegypti
237.5928	59.15850422	1.790306	0.697084845	0.313582509	0.804297	0.615154	0.946211659	0.896214754	Aedes Aegypti
237.5928	59.15850422	1.578713	0.730649831	0.281630166	0.810905	0.597808	0.949920631	0.899927884	Aedes Aegypti
237.5928	59.15850422	2.043122	0.654320776	0.351305662	0.800277	0.627748	0.940782731	0.893957601	Aedes Aegypti
235.3007	61.50382636	2.028409	0.674405343	0.364392857	0.774565	0.719603	0.935289013	0.879400541	Aedes Aegypti
235.3007	61.50382636	2.222856	0.647165396	0.402820234	0.767528	0.742074	0.928603852	0.875372748	Aedes Aegypti
235.3007	61.50382636	1.811461	0.709841632	0.336467898	0.777733	0.711752	0.938034127	0.881232082	Aedes Aegypti
235.3007	61.50382636	2.37865	0.622377292	0.419816173	0.765544	0.745704	0.927491541	0.874203654	Aedes Aegypti
236.6657	60.6870295	1.908729	0.688964324	0.332271942	0.797014	0.632518	0.943339076	0.892051712	Aedes Aegypti
236.6657	60.6870295	2.115437	0.659600335	0.364338078	0.79018	0.650745	0.939324933	0.888171338	Aedes Aegypti

236.6657	60.6870295	1.65323	0.731493228	0.292894849	0.800976	0.618369	0.949360898	0.894245621	Aedes Aegypti
236.6657	60.6870295	2.15673	0.652945575	0.371582713	0.78964	0.653076	0.937794421	0.887873434	Aedes Aegypti
232.8264	69.63926196	2.128339	0.714256157	0.341768983	0.781803	0.577677	0.94987394	0.88187016	Aedes Aegypti
232.8264	69.63926196	2.635887	0.649528232	0.416237383	0.770446	0.606534	0.940338448	0.875395213	Aedes Aegypti
232.8264	69.63926196	2.343837	0.685569081	0.373018965	0.777777	0.588325	0.945857795	0.879576013	Aedes Aegypti
232.8264	69.63926196	2.737446	0.636014073	0.43150679	0.768924	0.611791	0.93817909	0.87454838	Aedes Aegypti
236.767	61.10387488	1.790739	0.711292857	0.306685532	0.804072	0.602021	0.948605299	0.895914856	Aedes Aegypti
236.767	61.10387488	2.207599	0.64843709	0.371572766	0.79305	0.631971	0.939854891	0.889693081	Aedes Aegypti
236.767	61.10387488	1.540034	0.752638369	0.270165662	0.808303	0.589868	0.953226509	0.898283649	Aedes Aegypti
236.767	61.10387488	1.987296	0.683427283	0.335498506	0.797695	0.618823	0.945222342	0.892318336	Aedes Aegypti
236.4377	60.2777005	2.07062	0.653089178	0.371751366	0.784093	0.684524	0.935555767	0.88478991	Aedes Aegypti
236.4377	60.2777005	2.529019	0.581366323	0.441879264	0.772325	0.715249	0.926484777	0.878030979	Aedes Aegypti
236.4377	60.2777005	2.156721	0.639977251	0.385261109	0.781609	0.692254	0.933693134	0.883393913	Aedes Aegypti
236.4377	60.2777005	2.440117	0.595902655	0.430425472	0.773539	0.713133	0.927624644	0.878767027	Aedes Aegypti
236.9946	60.3336416	1.821618	0.699612496	0.316833279	0.8026	0.614119	0.94596997	0.895188449	Aedes Aegypti
236.9946	60.3336416	2.229773	0.636555076	0.379472732	0.791905	0.642169	0.937804948	0.889128914	Aedes Aegypti
236.9946	60.3336416	1.62142	0.733219225	0.285152848	0.80616	0.602312	0.950760984	0.897170454	Aedes Aegypti
236.9946	60.3336416	2.050757	0.665727409	0.35160586	0.795819	0.632387	0.941561269	0.891356737	Aedes Aegypti
236.7374	59.81866949	2.089911	0.644978915	0.373012388	0.786029	0.678585	0.936023505	0.885899594	Aedes Aegypti
236.7374	59.81866949	2.458859	0.586945977	0.429259489	0.777074	0.703019	0.928357388	0.880768908	Aedes Aegypti
236.7374	59.81866949	2.144763	0.636326524	0.38337923	0.78407	0.684448	0.934219621	0.884798437	Aedes Aegypti
236.7374	59.81866949	2.347156	0.605684832	0.413689688	0.778891	0.698796	0.930206075	0.88182744	Aedes Aegypti
236.5033	61.28059382	1.895267	0.690603351	0.326590971	0.79793	0.628252	0.945186439	0.892470062	Not Aedes Aegypti
236.5033	61.28059382	2.396281	0.613260898	0.406518726	0.784468	0.665032	0.934281302	0.884834444	Not Aedes Aegypti
236.5033	61.28059382	2.123064	0.65452653	0.36289012	0.791076	0.644829	0.940834682	0.888565736	Not Aedes Aegypti
236.5033	61.28059382	2.300907	0.62863104	0.391649839	0.786785	0.660124	0.936183854	0.886173135	Not Aedes Aegypti
238.3922	58.23597623	1.799452	0.675312692	0.313681046	0.813455	0.589234	0.947099823	0.901285151	Not Aedes Aegypti
238.3922	58.23597623	2.139086	0.618350627	0.364482674	0.804947	0.611423	0.940489871	0.896493996	Not Aedes Aegypti
238.3922	58.23597623	1.846594	0.667733269	0.320805915	0.812	0.592653	0.946225842	0.900462003	Not Aedes Aegypti
238.3922	58.23597623	2.138344	0.618258752	0.362025782	0.804637	0.611439	0.9414944	0.896330452	Not Aedes Aegypti
239.5319	56.34165665	1.453817	0.720978053	0.256249049	0.830244	0.539524	0.956123486	0.910619891	Not Aedes Aegypti

239.5319	56.34165665	1.878122	0.643786351	0.324288059	0.818721	0.573017	0.946590924	0.904255392	Not Aedes Aegypti
239.5319	56.34165665	1.684944	0.677432631	0.294731038	0.824469	0.557628	0.950408751	0.907449855	Not Aedes Aegypti
239.5319	56.34165665	1.865916	0.646131102	0.322858048	0.819463	0.572149	0.946523305	0.904678002	Not Aedes Aegypti
239.5223	55.53235605	1.656935	0.672749594	0.293995069	0.821697	0.581078	0.94894311	0.906039547	Not Aedes Aegypti
239.5223	55.53235605	1.899991	0.629075642	0.330345113	0.814597	0.600127	0.944413129	0.902081633	Not Aedes Aegypti
239.5223	55.53235605	1.649553	0.675257073	0.293274049	0.821537	0.582377	0.948853537	0.905958007	Not Aedes Aegypti
239.5223	55.53235605	1.876642	0.633785263	0.328566295	0.814983	0.600006	0.944259016	0.902294104	Not Aedes Aegypti
239.781	56.43690502	1.593183	0.692101818	0.273112188	0.832358	0.527263	0.954797915	0.91173417	Not Aedes Aegypti
239.781	56.43690502	1.977604	0.622153794	0.330066774	0.821923	0.55341	0.947986836	0.90593485	Not Aedes Aegypti
239.781	56.43690502	1.644195	0.682982436	0.282011986	0.830168	0.533092	0.953768028	0.910523553	Not Aedes Aegypti
239.781	56.43690502	2.119667	0.595013667	0.35374779	0.819401	0.562344	0.944186774	0.904552009	Not Aedes Aegypti
238.8962	57.50832539	1.523965	0.719661334	0.268364867	0.823945	0.556149	0.954127564	0.907109756	Not Aedes Aegypti
238.8962	57.50832539	1.875192	0.659043487	0.320718389	0.815155	0.580652	0.947459053	0.902218654	Not Aedes Aegypti
238.8962	57.50832539	1.644314	0.698387746	0.285278784	0.821728	0.562329	0.952041839	0.90587399	Not Aedes Aegypti
238.8962	57.50832539	2.00658	0.635074427	0.342200813	0.811869	0.588882	0.944427992	0.900384111	Not Aedes Aegypti
235.8096	62.61832072	2.006052	0.688168923	0.345085615	0.791075	0.639126	0.942671148	0.888477202	Not Aedes Aegypti
235.8096	62.61832072	2.315329	0.644257124	0.395365262	0.782346	0.664007	0.935442921	0.883540539	Not Aedes Aegypti
235.8096	62.61832072	1.955807	0.696482182	0.341739507	0.790781	0.640615	0.942473401	0.888329365	Not Aedes Aegypti
235.8096	62.61832072	2.354593	0.638151166	0.400349491	0.781261	0.66578	0.935016842	0.882911471	Not Aedes Aegypti
239.3042	56.4792928	1.528029	0.71034717	0.270678971	0.825526	0.557844	0.953204674	0.908071076	Not Aedes Aegypti
239.3042	56.4792928	1.845811	0.654063944	0.319369981	0.817411	0.581581	0.94661397	0.903567281	Not Aedes Aegypti
239.3042	56.4792928	1.515734	0.713309678	0.267602984	0.825575	0.55729	0.953988388	0.908086861	Not Aedes Aegypti
239.3042	56.4792928	1.91173	0.641715829	0.329324481	0.815949	0.584816	0.945414382	0.902741325	Not Aedes Aegypti
240.4692	54.90394042	1.667125	0.660612053	0.286611306	0.834325	0.525928	0.952816428	0.912878843	Not Aedes Aegypti
240.4692	54.90394042	1.910992	0.615249402	0.322120949	0.827652	0.542543	0.948485078	0.909178422	Not Aedes Aegypti
240.4692	54.90394042	1.633379	0.668060339	0.280927006	0.834274	0.524698	0.953975416	0.912839149	Not Aedes Aegypti
240.4692	54.90394042	2.077381	0.581709092	0.34911164	0.824118	0.552995	0.944696788	0.907221603	Not Aedes Aegypti
238.2512	58.5285456	1.798381	0.677848812	0.31070194	0.812748	0.588022	0.948269321	0.900852954	Not Aedes Aegypti
238.2512	58.5285456	2.072797	0.632545532	0.355864106	0.805538	0.610468	0.941445611	0.8968461	Not Aedes Aegypti
238.2512	58.5285456	1.809957	0.676227366	0.31253873	0.813159	0.588165	0.947696767	0.901087906	Not Aedes Aegypti
238.2512	58.5285456	2.033155	0.639622943	0.34861557	0.806738	0.607062	0.942456541	0.897516076	Not Aedes Aegypti

239.9016	56.11668297	1.541736	0.699281681	0.263741923	0.834089	0.523814	0.956274211	0.912706086	Not Aedes Aegypti
239.9016	56.11668297	1.951584	0.623420928	0.326628601	0.822889	0.553326	0.948314895	0.906501379	Not Aedes Aegypti
239.9016	56.11668297	1.598106	0.688821156	0.275189796	0.831771	0.531266	0.954465986	0.911439803	Not Aedes Aegypti
239.9016	56.11668297	2.067002	0.601173929	0.34621199	0.820926	0.560612	0.944983127	0.905428415	Not Aedes Aegypti
238.9387	57.3227056	1.778555	0.669722751	0.305840378	0.819677	0.571049	0.949000974	0.904767334	Not Aedes Aegypti
238.9387	57.3227056	2.186958	0.598583637	0.3706184	0.808521	0.601419	0.940256073	0.898539075	Not Aedes Aegypti
238.9387	57.3227056	1.979586	0.633349361	0.337073042	0.813954	0.585201	0.94534415	0.90156462	Not Aedes Aegypti
238.9387	57.3227056	2.102227	0.614132764	0.356404572	0.810693	0.59635	0.942248562	0.899764134	Not Aedes Aegypti
240.0153	55.65363579	1.777167	0.648077311	0.305908062	0.828045	0.544663	0.949771704	0.909413651	Not Aedes Aegypti
240.0153	55.65363579	1.915921	0.625009938	0.328109236	0.823741	0.557678	0.946493709	0.907044136	Not Aedes Aegypti
240.0153	55.65363579	1.622745	0.67952184	0.281940213	0.830871	0.536099	0.952943621	0.910967974	Not Aedes Aegypti
240.0153	55.65363579	2.034759	0.601719878	0.34426666	0.820704	0.563057	0.945056308	0.905327917	Not Aedes Aegypti
237.5518	58.85220505	2.162448	0.623436488	0.376568545	0.795865	0.648554	0.936432504	0.891488449	Not Aedes Aegypti
237.5518	58.85220505	2.388761	0.590296742	0.412564339	0.788375	0.668959	0.9316066	0.887245801	Not Aedes Aegypti
237.5518	58.85220505	1.979577	0.656954358	0.346184778	0.799212	0.638794	0.941051285	0.893376706	Not Aedes Aegypti
237.5518	58.85220505	2.257667	0.612833311	0.393014233	0.791122	0.662686	0.934003963	0.888819108	Not Aedes Aegypti
243.2379	49.06250403	1.1921	0.698796362	0.214711343	0.864941	0.45299	0.96216026	0.929761765	Not Aedes Aegypti
243.2379	49.06250403	1.430832	0.643508995	0.251216924	0.85781	0.472691	0.957472284	0.925891889	Not Aedes Aegypti
243.2379	49.06250403	1.293047	0.674783111	0.231072168	0.86129	0.462711	0.960169314	0.927785708	Not Aedes Aegypti
243.2379	49.06250403	1.512272	0.622941073	0.264201645	0.855324	0.479006	0.955965712	0.924545506	Not Aedes Aegypti
238.6551	57.12933639	2.130139	0.605985512	0.369514749	0.805676	0.619323	0.938256901	0.89703646	Not Aedes Aegypti
238.6551	57.12933639	2.294241	0.582462685	0.39515412	0.799687	0.635201	0.934909123	0.89365949	Not Aedes Aegypti
238.6551	57.12933639	1.988092	0.634458323	0.34692639	0.807762	0.612925	0.941674264	0.898196907	Not Aedes Aegypti
238.6551	57.12933639	2.234784	0.593412623	0.388190939	0.80099	0.633263	0.935303076	0.894405311	Not Aedes Aegypti
239.8297	56.09776946	1.550483	0.69741125	0.271942848	0.831371	0.534917	0.954092072	0.911241533	Not Aedes Aegypti
239.8297	56.09776946	1.980135	0.618286563	0.338141771	0.820874	0.564052	0.945140422	0.905429897	Not Aedes Aegypti
239.8297	56.09776946	1.529522	0.70253418	0.268502182	0.831452	0.533996	0.95455536	0.911272633	Not Aedes Aegypti
239.8297	56.09776946	1.833921	0.646469055	0.314022593	0.823708	0.554986	0.948804868	0.906988451	Not Aedes Aegypti
236.8101	60.46740437	1.774892	0.701539589	0.314417035	0.799264	0.632996	0.945790652	0.893306551	Not Aedes Aegypti
236.8101	60.46740437	2.228028	0.62930721	0.385581623	0.786216	0.666777	0.936784938	0.885906204	Not Aedes Aegypti
236.8101	60.46740437	1.928308	0.676489044	0.338677902	0.794301	0.645288	0.943168939	0.890494935	Not Aedes Aegypti

236.8101	60.46740437	2.25664	0.624587683	0.389182455	0.786932	0.667719	0.935802261	0.886329991	Not Aedes Aegypti
240.0527	55.82749999	1.843437	0.635310694	0.313031475	0.828277	0.537011	0.94970084	0.909475226	Not Aedes Aegypti
240.0527	55.82749999	1.958009	0.617234965	0.33040767	0.824753	0.547688	0.94722058	0.907534693	Not Aedes Aegypti
240.0527	55.82749999	1.685407	0.667463272	0.28836836	0.830925	0.528698	0.953150472	0.910932906	Not Aedes Aegypti
240.0527	55.82749999	2.0101	0.606978809	0.338189693	0.822774	0.55109	0.946640476	0.906424827	Not Aedes Aegypti
237.3121	59.82681697	1.639592	0.719631668	0.290015369	0.806564	0.604956	0.950426786	0.89741558	Not Aedes Aegypti
237.3121	59.82681697	2.100193	0.644881166	0.362382184	0.795013	0.638776	0.940351893	0.890923294	Not Aedes Aegypti
237.3121	59.82681697	1.830002	0.687892409	0.321215908	0.802473	0.618493	0.945653073	0.895123935	Not Aedes Aegypti
237.3121	59.82681697	1.859538	0.685704258	0.328159939	0.80037	0.625711	0.944067157	0.893956613	Not Aedes Aegypti
240	56.12755273	1.6226	0.682402619	0.275261902	0.834578	0.517931	0.955091046	0.912954766	Not Aedes Aegypti
240	56.12755273	1.985175	0.615604884	0.328602281	0.825143	0.541511	0.948631337	0.907717659	Not Aedes Aegypti
240	56.12755273	1.645875	0.67861204	0.279640141	0.833105	0.521606	0.954675805	0.912138347	Not Aedes Aegypti
240	56.12755273	2.140012	0.585623674	0.35407869	0.822107	0.55117	0.944742341	0.906043361	Not Aedes Aegypti
237.3654	59.42566654	1.661363	0.711263205	0.300720426	0.802748	0.626513	0.947220736	0.895327255	Not Aedes Aegypti
237.3654	59.42566654	1.958837	0.663420376	0.349467502	0.795116	0.651491	0.93976944	0.891056608	Not Aedes Aegypti
237.3654	59.42566654	1.57723	0.726604248	0.289411634	0.804252	0.622169	0.948357794	0.896158231	Not Aedes Aegypti
237.3654	59.42566654	1.87588	0.677695991	0.337192568	0.796104	0.647155	0.941566071	0.891602987	Not Aedes Aegypti
238.7998	57.91041288	1.612319	0.705787105	0.278087013	0.823028	0.554523	0.953531743	0.906550519	Not Aedes Aegypti
238.7998	57.91041288	2.067421	0.62694657	0.349518943	0.810976	0.587827	0.943937637	0.899846385	Not Aedes Aegypti
238.7998	57.91041288	1.749706	0.681294936	0.302202093	0.819182	0.566712	0.9498665	0.904435868	Not Aedes Aegypti
238.7998	57.91041288	2.128388	0.615958814	0.359540266	0.810108	0.591292	0.942254042	0.899372401	Not Aedes Aegypti
237.618	59.52310514	1.752972	0.697065693	0.305334441	0.808784	0.59975	0.948514791	0.898608618	Not Aedes Aegypti
237.618	59.52310514	2.23363	0.618415824	0.380379354	0.797023	0.633037	0.938089296	0.892001527	Not Aedes Aegypti
237.618	59.52310514	2.044027	0.648181703	0.3499537	0.801728	0.618502	0.942629328	0.894633643	Not Aedes Aegypti
237.618	59.52310514	2.091239	0.642695975	0.357664637	0.799265	0.626126	0.941582969	0.893266892	Not Aedes Aegypti
237.5986	59.44906778	1.812787	0.689018861	0.317152553	0.806811	0.605836	0.945997994	0.897557284	Not Aedes Aegypti
237.5986	59.44906778	2.180538	0.630342474	0.37613758	0.796979	0.633535	0.93770361	0.892037093	Not Aedes Aegypti
237.5986	59.44906778	1.848136	0.683651109	0.325188437	0.805331	0.610771	0.944528116	0.896739137	Not Aedes Aegypti
237.5986	59.44906778	2.010536	0.659247159	0.347675794	0.800805	0.622396	0.941899249	0.894177268	Not Aedes Aegypti
237.7672	59.62989671	1.547545	0.733833673	0.271178136	0.81587	0.573027	0.953649164	0.902508007	Not Aedes Aegypti
237.7672	59.62989671	1.917372	0.673852563	0.326276553	0.80686	0.598313	0.946639526	0.897473293	Not Aedes Aegypti

237.7672	59.62989671	1.69136	0.709828615	0.292606525	0.812927	0.581085	0.950822345	0.900865429	Not Aedes Aegypti
237.7672	59.62989671	2.043224	0.652331821	0.346908788	0.803612	0.606826	0.943737652	0.895656347	Not Aedes Aegypti
239.1685	56.1584215	1.81502	0.65175917	0.320529676	0.815734	0.595641	0.944798062	0.902704771	Not Aedes Aegypti
239.1685	56.1584215	2.10043	0.601850121	0.366226326	0.806786	0.61899	0.938789187	0.897690029	Not Aedes Aegypti
239.1685	56.1584215	1.598184	0.694353105	0.286679864	0.8197	0.583488	0.949668133	0.904909927	Not Aedes Aegypti
239.1685	56.1584215	1.875237	0.644504734	0.331620044	0.812079	0.606352	0.943136884	0.900679576	Not Aedes Aegypti
237.8313	59.21005014	1.687381	0.706061419	0.29298873	0.812374	0.586908	0.950738091	0.900607123	Not Aedes Aegypti
237.8313	59.21005014	2.102426	0.637994348	0.355753504	0.802616	0.614772	0.942340514	0.895145531	Not Aedes Aegypti
237.8313	59.21005014	1.947399	0.661638644	0.333057137	0.80621	0.60324	0.945449603	0.897147065	Not Aedes Aegypti
237.8313	59.21005014	2.02929	0.650492221	0.345806369	0.802603	0.61297	0.943964517	0.89512579	Not Aedes Aegypti
238.365	58.32750865	1.735362	0.690757799	0.303699611	0.81556	0.580477	0.948238249	0.902464901	Not Aedes Aegypti
238.365	58.32750865	1.957508	0.655340549	0.337256453	0.808897	0.598579	0.944108795	0.89873634	Not Aedes Aegypti
238.365	58.32750865	1.708319	0.696519197	0.302429854	0.815224	0.58167	0.947938239	0.90228215	Not Aedes Aegypti
238.365	58.32750865	2.058543	0.637516774	0.35436037	0.806827	0.605121	0.941287909	0.897593116	Not Aedes Aegypti
239.6106	56.00205785	1.809074	0.64718243	0.30877039	0.823928	0.561604	0.949135207	0.907176962	Not Aedes Aegypti
239.6106	56.00205785	2.230493	0.569996223	0.375162927	0.812604	0.591626	0.940230779	0.900860395	Not Aedes Aegypti
239.6106	56.00205785	2.011562	0.60891865	0.339963082	0.818118	0.575299	0.945483568	0.903933778	Not Aedes Aegypti
239.6106	56.00205785	2.11545	0.592162197	0.356929259	0.815278	0.585563	0.942657232	0.90237053	Not Aedes Aegypti
239.7521	55.83565461	1.744803	0.656351589	0.300610755	0.825745	0.555166	0.950058392	0.908165695	Not Aedes Aegypti
239.7521	55.83565461	2.067069	0.597177492	0.348041003	0.816483	0.577736	0.944704056	0.903003464	Not Aedes Aegypti
239.7521	55.83565461	1.779497	0.65047813	0.306396072	0.824039	0.559877	0.949422304	0.907228107	Not Aedes Aegypti
239.7521	55.83565461	2.1043	0.59009032	0.355418821	0.816915	0.579321	0.942732029	0.90324613	Not Aedes Aegypti
238.9292	57.76544731	2.036479	0.624405365	0.343457861	0.815676	0.569517	0.945197776	0.902408476	Not Aedes Aegypti
238.9292	57.76544731	2.118804	0.614025586	0.354163841	0.812521	0.577679	0.944069136	0.900644293	Not Aedes Aegypti
238.9292	57.76544731	1.859687	0.657849646	0.316369048	0.818054	0.561453	0.949159638	0.903726609	Not Aedes Aegypti
238.9292	57.76544731	2.244831	0.591026949	0.37547356	0.809232	0.586491	0.941130453	0.89880702	Not Aedes Aegypti
238.5589	54.30275047	1.253535	0.744794743	0.25231992	0.804497	0.638625	0.949567174	0.89651726	Not Aedes Aegypti
238.5589	54.30275047	1.388538	0.720847372	0.275932541	0.799277	0.655134	0.945656181	0.893591405	Not Aedes Aegypti
238.5589	54.30275047	1.215361	0.753417244	0.245996805	0.803697	0.637633	0.951114205	0.896051327	Not Aedes Aegypti
238.5589	54.30275047	1.506342	0.697115281	0.29382617	0.796459	0.662947	0.943509604	0.891998857	Not Aedes Aegypti
238.6355	54.40627849	1.366611	0.722565936	0.268933656	0.804623	0.639801	0.947435848	0.896600805	Not Aedes Aegypti

238.6355	54.40627849	1.486321	0.701857951	0.288120359	0.800395	0.65292	0.944514698	0.894230672	Not Aedes Aegypti
238.6355	54.40627849	1.33853	0.729035139	0.261773246	0.804318	0.637727	0.949529121	0.896414072	Not Aedes Aegypti
238.6355	54.40627849	1.617921	0.6754268	0.309192832	0.7971	0.662024	0.941654298	0.89236879	Not Aedes Aegypti
240.6038	51.81026694	1.465676	0.674080842	0.27186484	0.824126	0.57511	0.950985051	0.907435084	Not Aedes Aegypti
240.6038	51.81026694	1.517181	0.666582147	0.280443684	0.821248	0.584359	0.949631834	0.905851776	Not Aedes Aegypti
240.6038	51.81026694	1.211652	0.731113177	0.230064694	0.829504	0.557929	0.957198834	0.910413322	Not Aedes Aegypti
240.6038	51.81026694	1.502999	0.669849312	0.280039767	0.821387	0.584392	0.949220565	0.905927106	Not Aedes Aegypti
239.2773	53.48232512	1.194116	0.750863377	0.233769276	0.814435	0.599735	0.955028841	0.90200955	Not Aedes Aegypti
239.2773	53.48232512	1.39492	0.712361177	0.263245879	0.807871	0.616718	0.951722438	0.89832915	Not Aedes Aegypti
239.2773	53.48232512	1.194295	0.75125331	0.232252365	0.814513	0.599086	0.955531456	0.902048618	Not Aedes Aegypti
239.2773	53.48232512	1.459346	0.699261927	0.280119052	0.805691	0.627024	0.947653566	0.897131876	Not Aedes Aegypti
240.7125	50.78767289	1.044584	0.756576437	0.203896487	0.829784	0.555857	0.960559397	0.910550403	Not Aedes Aegypti
240.7125	50.78767289	1.273581	0.706492812	0.244283767	0.821808	0.582371	0.954171091	0.906162472	Not Aedes Aegypti
240.7125	50.78767289	1.085326	0.747814349	0.214871068	0.827523	0.566015	0.958042922	0.90932896	Not Aedes Aegypti
240.7125	50.78767289	1.393722	0.678922434	0.267535414	0.818193	0.594051	0.950099254	0.904156032	Not Aedes Aegypti
239.8699	51.36018368	1.381926	0.685102895	0.274701986	0.80788	0.630132	0.947049487	0.898422316	Not Aedes Aegypti
239.8699	51.36018368	1.690904	0.619277194	0.33124748	0.79722	0.662906	0.937984633	0.892441931	Not Aedes Aegypti
239.8699	51.36018368	1.653774	0.624264843	0.322461402	0.800138	0.653478	0.939811669	0.894074739	Not Aedes Aegypti
239.8699	51.36018368	1.717406	0.613410069	0.334159934	0.796819	0.662675	0.937955185	0.892202938	Not Aedes Aegypti
242.0757	47.67111645	1.268628	0.663931833	0.254197936	0.831309	0.567232	0.950741645	0.911488918	Not Aedes Aegypti
242.0757	47.67111645	1.559719	0.591495555	0.304647421	0.821304	0.59506	0.943354098	0.905945939	Not Aedes Aegypti
242.0757	47.67111645	1.403366	0.62940228	0.277888167	0.826942	0.579677	0.947197923	0.909073633	Not Aedes Aegypti
242.0757	47.67111645	1.397894	0.634006659	0.275347693	0.826314	0.58098	0.947840409	0.908723678	Not Aedes Aegypti
241.1834	49.17317886	1.288887	0.680353934	0.256941626	0.821883	0.585649	0.95056443	0.906220493	Not Aedes Aegypti
241.1834	49.17317886	1.509466	0.629909296	0.29473239	0.814287	0.60626	0.944908732	0.901988818	Not Aedes Aegypti
241.1834	49.17317886	1.203493	0.702203605	0.238406311	0.824993	0.574304	0.95384516	0.907935352	Not Aedes Aegypti
241.1834	49.17317886	1.359334	0.666632459	0.26841326	0.818611	0.595272	0.948857938	0.904411443	Not Aedes Aegypti
238.6729	53.9629884	1.483545	0.693712985	0.288221421	0.799708	0.653619	0.945352882	0.893827512	Not Aedes Aegypti
238.6729	53.9629884	1.79436	0.634135153	0.339038264	0.790836	0.679273	0.937971299	0.88880741	Not Aedes Aegypti
238.6729	53.9629884	1.801633	0.629262739	0.33933143	0.792356	0.674425	0.938062981	0.889662352	Not Aedes Aegypti
238.6729	53.9629884	1.92883	0.606552072	0.363525846	0.786714	0.691488	0.934159661	0.886474838	Not Aedes Aegypti

237.6723	62.10197783	1.302804	0.780178883	0.217690303	0.831503	0.466827	0.966645237	0.910463982	Not Aedes Aegypti
237.6723	62.10197783	1.888999	0.685276438	0.304735404	0.817785	0.504284	0.955433065	0.902946346	Not Aedes Aegypti
237.6723	62.10197783	1.510235	0.746297423	0.246844953	0.827245	0.479672	0.96308096	0.908158355	Not Aedes Aegypti
237.6723	62.10197783	1.471168	0.754827446	0.242229795	0.826359	0.481298	0.963594245	0.907640549	Not Aedes Aegypti
237.759	55.32883132	1.26941	0.7517666	0.25935058	0.794953	0.667721	0.947354557	0.891145184	Not Aedes Aegypti
237.759	55.32883132	1.519694	0.706251221	0.300569663	0.786451	0.692182	0.941733328	0.886322129	Not Aedes Aegypti
237.759	55.32883132	1.242624	0.757547053	0.255201032	0.794843	0.66802	0.948154714	0.891086038	Not Aedes Aegypti
237.759	55.32883132	1.561915	0.698067733	0.30936336	0.785791	0.69716	0.939764993	0.885968521	Not Aedes Aegypti
240.893	52.13348979	1.247756	0.725393959	0.231600612	0.836223	0.53295	0.957993182	0.914083885	Not Aedes Aegypti
240.893	52.13348979	1.570669	0.658401286	0.282990779	0.8271	0.559716	0.950786029	0.909059419	Not Aedes Aegypti
240.893	52.13348979	1.266364	0.722286394	0.230847497	0.83632	0.532008	0.958726826	0.914129083	Not Aedes Aegypti
240.893	52.13348979	1.343677	0.707810267	0.245053049	0.832894	0.542796	0.956373643	0.912253837	Not Aedes Aegypti
238.3985	54.15212374	1.134892	0.766409846	0.236619525	0.802213	0.647335	0.951004291	0.895247379	Not Aedes Aegypti
238.3985	54.15212374	1.617719	0.670705605	0.315611031	0.788939	0.68873	0.939556381	0.887770322	Not Aedes Aegypti
238.3985	54.15212374	1.460296	0.700313024	0.28738356	0.794847	0.671301	0.944026112	0.891108495	Not Aedes Aegypti
238.3985	54.15212374	1.689038	0.65628132	0.328658551	0.78698	0.693862	0.937468555	0.886647758	Not Aedes Aegypti
239.3744	51.14062043	1.486052	0.657578891	0.3091305	0.790948	0.677047	0.939145335	0.888855672	Not Aedes Aegypti
239.3744	51.14062043	1.834571	0.581938396	0.374399997	0.778555	0.712457	0.928670245	0.881799558	Not Aedes Aegypti
239.3744	51.14062043	1.70329	0.608560055	0.348827649	0.784165	0.696518	0.932892064	0.885002463	Not Aedes Aegypti
239.3744	51.14062043	1.71846	0.608699544	0.350977097	0.782766	0.70007	0.932585013	0.884189409	Not Aedes Aegypti
238.6052	52.64757104	1.457359	0.681670411	0.303222025	0.786867	0.69139	0.939672013	0.886566837	Not Aedes Aegypti
238.6052	52.64757104	1.910075	0.587479762	0.381243346	0.772784	0.729279	0.927975769	0.878506494	Not Aedes Aegypti
238.6052	52.64757104	1.557913	0.660914605	0.318239255	0.784414	0.69669	0.937609751	0.885153643	Not Aedes Aegypti
238.6052	52.64757104	1.594031	0.655938155	0.322007082	0.782561	0.700871	0.937309359	0.884093883	Not Aedes Aegypti
237.8621	53.94270387	1.869073	0.613574896	0.368353695	0.772859	0.730964	0.930952963	0.878551187	Not Aedes Aegypti
237.8621	53.94270387	2.04761	0.581467088	0.402321773	0.765668	0.751835	0.925232582	0.874423932	Not Aedes Aegypti
237.8621	53.94270387	1.743385	0.640947748	0.344651153	0.776336	0.719333	0.934556411	0.880520878	Not Aedes Aegypti
237.8621	53.94270387	1.88179	0.615475613	0.372784552	0.770516	0.738434	0.92949563	0.877211867	Not Aedes Aegypti
240.1709	49.85717871	1.550303	0.621574009	0.316584461	0.79675	0.65971	0.939112961	0.892138645	Not Aedes Aegypti
240.1709	49.85717871	1.459284	0.647850751	0.300745623	0.797763	0.65744	0.941409121	0.892720694	Not Aedes Aegypti
240.1709	49.85717871	1.527573	0.628601853	0.312701179	0.796879	0.659123	0.939692808	0.892214437	Not Aedes Aegypti

240.1709	49.85717871	1.881352	0.545784738	0.37765631	0.784615	0.690752	0.929583568	0.885223317	Not Aedes Aegypti
239.6284	51.82573095	1.406675	0.681947218	0.282393755	0.803866	0.644673	0.945379409	0.896189799	Not Aedes Aegypti
239.6284	51.82573095	1.861383	0.58319574	0.359954286	0.790438	0.68173	0.933973736	0.888587571	Not Aedes Aegypti
239.6284	51.82573095	1.77799	0.598551731	0.344383795	0.794259	0.67119	0.936336351	0.890756596	Not Aedes Aegypti
239.6284	51.82573095	1.8722	0.580652964	0.362931228	0.79001	0.684132	0.933170625	0.888357513	Not Aedes Aegypti
238.7721	53.99900288	1.358825	0.720375526	0.270078274	0.804085	0.642089	0.947111965	0.896300669	Not Aedes Aegypti
238.7721	53.99900288	1.519553	0.691004867	0.300487517	0.797628	0.662648	0.941865438	0.892680861	Not Aedes Aegypti
238.7721	53.99900288	1.382048	0.716429627	0.273146613	0.80315	0.644457	0.946825095	0.895773938	Not Aedes Aegypti
238.7721	53.99900288	1.539725	0.686838138	0.297889201	0.797852	0.659351	0.943756688	0.892789633	Not Aedes Aegypti